



**Kincir Air Alternatif Dengan Timer Sebagai Penyuplai  
Kandungan Oksigen (*Dissolved Oxygen*) Pada Kolam Pembenihan Lele  
Berbasis Mikrokontroler ATmega8**

**LAPORAN PROYEK AKHIR**

Diajukan Pada Fakultas Teknik Universitas Negeri Yogyakarta  
Untuk Memenuhi Sebagian Persyaratan  
Guna Memperoleh Gelar Ahli Madya



**Oleh :**

**SRI ANGGANA ANGGAKARA**

**NIM. 08506134003**

**PROGRAM STUDI TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
2012**

## PERSETUJUAN

Proyek akhir yang berjudul “Kincir Air Alternatif Dengan Timer Sebagai Penyuplai Kandungan Oksigen (*Dissolved Oxygen*) Pada Kolam Pembenihan Lele Berbasis Mikrokontroler Atmega8” ini telah disetujui oleh pembimbing untuk diujikan.



Yogyakarta, 10 Agustus 2012

Dosen Pembimbing,

**Muhamad Ali, M.T.**  
**NIP 19741127 200003 1 005**

## PENGESAHAN

Proyek akhir yang berjudul “Kincir Air Alternatif Dengan Timer Sebagai Penyuplai Kandungan Oksigen (*Dissolved Oxygen*) Pada Kolam Pembenihan Ikan Lele Berbasis Mikrokontroler Atmega8” ini telah dipertahankan di depan Dewan Penguji pada tanggal 14 Agustus 2012 dan dinyatakan lulus.

## DEWAN PENGUJI

Nama	Jabatan	Tandatangan	Tanggal
Muhamad Ali, M.T.	Ketua Penguji		20-8-2012
Herlambang Sigit P, S.T., M.Cs.	Sekretaris Penguji		20/8/12
Moh. Khairudin, M.T., Ph.D	Penguji		20/8/2012

Yogyakarta, 14 Agustus 2012

Fakultas Teknik

Universitas Negeri Yogyakarta

Dekan,



**Dr. Moch. Bruri Triyono**

**NIP. 19560216 198603 1 003**

## **SURAT PERNYATAAN**

Dengan ini saya menyatakan bahwa dalam Proyek Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Ahli Madya atau gelar lainnya di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis oleh orang lain, kecuali secara tertulis yang diacu dalam naskah ini dan disebutkan dalam daftar pustaka.



Yogyakarta, 10 Agustus 2012

Yang menyatakan,



**Sri Anggana Anggakara**

**NIM. 08506134003**

**“KINCIR AIR ALTERNATIF DENGAN TIMER SEBAGAI PENYUPLAI  
KANDUNGAN OKSIGEN (*DISSOLVED OXYGEN*) PADA KOLAM  
PEMBENIHAN LELE BERBASIS MIKROKONTROLER ATMEGA8”**

Oleh:  
SRI ANGGANA ANGGAKARA  
NIM: 08506134003

**ABSTRAK**

Pembuatan Proyek Akhir yang berjudul Kincir Air Alternatif Dengan Timer Sebagai Penyuplai Kandungan Oksigen (*Dissolved Oxygen*) Pada Kolam Pembenihan Lele Berbasis Mikrokontroler ATmega8 bertujuan untuk memberikan suplai kandungan oksigen pada kolam bibit ikan lele pada malam hari yang disebabkan tidak bisanya fitoplankton berfotosintesis karena kurangnya cahaya matahari yang berfungsi untuk membantu fitoplankton berfotosintesis.

Metode yang digunakan dalam proyek akhir ini adalah pembuatan sistem kincir air otomatis dengan timer berbasis mikrokontroler yang dapat diatur waktu hidup dan mati kincir air tersebut berbasis mikrokontroler ATmega8. Langkah – langkah pembuatan proyek akhir ini antara lain membuat konsep rancangan perangkat keras, membuat konsep rancangan perangkat lunak, analisis kebutuhan, identifikasi alat dan bahan, perancangan perangkat keras, perancangan perangkat lunak dan juga pengoperasian dan pengujian. Perancangan perangkat keras ini terdiri dari beberapa bagian yaitu : catu daya, pengolah data dan perangkat kontrol yang berupa sistem minimum ATmega8, tombol seting, saklar relay dan juga *LCD* monitor. Tombol seting ini berfungsi untuk mengatur seting waktu , waktu hidup dan mati kincir yang nantinya akan diproses pada perangkat kontrol dan pengolah data yaitu sistem minimum ATmega8 dan akan ditampilkan pada *LCD* monitor. Hasil pengolahan data dari sistem minimum ATmega8 akan dikirim ke saklar relay guna mengontrol kincir air yang digerakkan oleh motor DC power window.

Berdasarkan dari hasil dan unjuk kerja pengujian “Kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen pada kolam pembenihan ikan lele berbasis mikrokontroler ATmega8” telah menunjukkan hasil yang sesuai dengan perencanaan. *Kincir air dapat bekerja apabila saklar relay mendapat logika 1 dari output mikrokontroler yang menjadi inputan untuk saklar relay. saklar relay akan menghidupkan motor dc dengan kecepatan rata – rata 100 rpm untuk memutar kincir air. Ketepatan proses alat dengan kebenaran 97.4%. Alat dapat bekerja mulai dari proses seting waktu, proses seting kincir on dan proses seting kincir off. Semua akan berjalan otomatis setelah proses seting waktu ditentukan.*

**Kata kunci :** Timer, LCD, Mikrokontroler ATmega8.

## *MOTTO*



*“BANGUN DARI MIMPI DAN WUJUDKAN MIMPI MIMPIMU ITU”*

*“JANGAN PERNAH MENYERAH SEBELUM BERTANDING”*

*“SEMANGAT PANTANG MUNDUR”*

## **PERSEMBAHAN**

Tugas akhir ini saya persembahkan untuk :

- ☺ Allah SWT yang terus membimbing hambaNYA dimanapun, disaat apapun dan dalam kondisi apapun.
- ☺ Kepada keluarga saya yang tiada hentinya mendoakan dan memberikan semangat disaat anaknya mendapatkan masalah dalam pembuatan tugas akhir ini.
- ☺ Yahya Hamitisna S.Pi , Buyung Purnomo W. S.Pi. yang member ide dan penyemangat serta Sri Nuryanti yang membuat saya untuk tetap semangat dalam menjalani hidup.
- ☺ Teman–teman kelas C 2008, teman yang ada disaat apapun, susah senang bersama yang tidak akan pernah terlupakan. Dan teman-teman lainnya yang tidak bisa disebutkan satu persatu.
- ☺ Almamaterku Universitas Negeri Yogyakarta Fakultas Teknik Jurusan Pendidikan Teknik Elektro.

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat, karunia dan nikmat yang telah diberikan-Nya, sehingga Penulis dapat menyelesaikan proyek akhir dan penyusunan laporan ini.

Penulis sadar tanpa bantuan berbagai pihak Proyek Akhir ini tidak akan terlaksana dengan baik. Oleh karena itu pada kesempatan ini penulis dengan ketulusan hati mengucapkan terima kasih atas dukungan, bimbingan dan bantuannya baik secara moril maupun materil kepada :

1. Allah SWT yang telah melimpahkan rahmat dan nikmat islam.
2. Bapak Prof.Dr. Rochmat Wahab, MA selaku Rektor Universitas Negeri Yogyakarta.
3. Bapak Dr.Moch. Bruri Triyono, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
4. Bapak Drs. Ketut Ima Ismara, M.Pd., M.Kes. selaku kepala jurusan Pendidikan Teknik Elektro Fakultas Teknik Universitas Negeri Yogyakarta.
5. Bapak Muhamad Ali, MT. selaku Dosen pembimbing Proyek Akhir Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Yogyakarta.
6. Bapak Drs. Nyoman Astra. selaku Penasehat Akademik.
7. Kedua orang tua, kakakku, dan adikku yang telah memberikan semangat untuk mengerjakan Proyek Akhir ini.
8. Bapak/Ibu Dosen Elektro FT UNY yang telah membantu dalam penyusunan laporan Proyek Akhir.



9. Teman kelas C angkatan 2008 semua yang telah memberikan semangat dan bantuan dalam penyusunan laporan Proyek Akhir ini.
10. Bapak Mardiyanto selaku staf TU Jurusan Teknik Elektro Universitas Negeri Yogyakarta yang telah memberikan bantuan dan pelayanan dengan baik.
11. Semua pihak yang telah membantu, serta yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa Laporan Proyek Akhir ini masih banyak kekurangannya, oleh karena itu saran dan kritik yang membangun sangat penulis harapkan demi kesempurnaan di masa yang akan datang. Akhirnya Penulis berharap semoga Laporan Proyek Akhir ini dapat bermanfaat bagi diri Penulis dan pembaca semuanya.

Yogyakarta, Agustus 2012

Penulis

## DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PERSETUJUAN</b> .....	ii
<b>LEMBAR PENGESAHAN</b> .....	iii
<b>LEMBAR PERNYATAAN KEASLIAN</b> .....	iv
<b>ABSTRAK</b> .....	v
<b>MOTTO</b> .....	vi
<b>PERSEMBAHAN</b> .....	vii
<b>KATA PENGANTAR</b> .....	viii
<b>DAFTAR ISI</b> .....	x
<b>DAFTAR GAMBAR</b> .....	xiii
<b>DAFTAR TABEL</b> .....	xv
<b>DAFTAR LAMPIRAN</b> .....	xvi
 <b>BAB I    PENDAHULUAN</b> .....	 1
A. Latar Belakang.....	1
B. Identifikasi Masalah.....	3
C. Batasan Masalah.....	3
D. Rumusan Masalah.....	3
E. Tujuan.....	4
F. Manfaat.....	4
G. Keaslian gagasan.....	5
 <b>BAB II    PENDEKATAN PEMECAHAN MASALAH</b> .....	 6
A. Motor DC .....	8
1. Prinsip dasar dan cara kerja motor DC.....	10
2. Prinsip arah putaran motor.....	12
3. Electromotive Force / Gaya Gerak Listrik.....	13
B. Dioda .....	14

C. Catu Daya .....	15
D. Push Button .....	18
E. Relay .....	19
F. Mikrokontroler ATmega8.....	20
1. Spesifikasi dari mikokontroler ATmega8.....	21
2. Konfigurasi dari fungsi kaki pin ATmega8.....	21
G. LCD( <i>Liquid Cristal Display</i> ) .....	22
H. Resistor .....	23
I. Transistor.....	23
J. Diagram alir( <i>Flowchart</i> ) .....	27
 <b>BAB III KONSEP RANCANGAN ALAT</b> .....	29
A. Konsep Rancangan Perangkat Keras ( <i>hardware</i> ) .....	29
B. Identifikasi Alat dan Bahan yang Dibutuhkan .....	31
C. Perancangan Perangkat Keras.....	35
1. Catu Daya ( <i>Power Supply</i> ).....	35
2. Perangkat Kontrol dan Pengolah Data .....	38
3. LCD Monitor.....	40
4. Tombol <i>Setting</i> .....	40
5. Saklar Relay .....	41
6. Pembuatan Jalur PCB dan Perakitan Komponen .....	42
7. Pembuatan Boks .....	43
D. Perancangan Sistem Mekanis .....	46
a. Perancangan konstruksi kincir air.....	46
b. Perancangan as penghubung motor dengan kincir.....	48
c. Desain mekanis secara keseluruhan.....	48
E. Perancangan Perangkat Lunak .....	49
F. Rencana Pengoperasian dan Pengujian Data.....	58
1. Alat yang digunakan untuk pengujian dan pengambilan data.....	59
2. Lokasi pengujian dan pengambilan data.....	59

3. Prosedur pengoperasian alat.....	60
4. Pengujian Alat Meliputi.....	62
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>65</b>
A. Pengujian .....	65
1. Tujuan .....	65
2. Tempat dan waktu pengambilan data.....	65
B. Hasil pengujian alat.....	65
1. Catu Daya ( <i>Power Supply</i> ).....	65
2. Sistem <i>Driver Relay</i> .....	69
3. Tombol <i>Setting</i> .....	70
4. Pengujian Alat.....	71
C. Pembahasan .....	72
1. Catu Daya .....	72
2. Unjuk kerja sistem.....	75
<b>BAB V KESIMPULAN DAN SARAN.....</b>	<b>77</b>
A. Kesimpulan.....	77
B. Keterbatasan Alat .....	78
C. Saran.....	79
<b>DAFTAR PUSTAKA .....</b>	<b>80</b>
<b>LAMPIRAN.....</b>	<b>81</b>

## DAFTAR GAMBAR

	Halaman
Gambar 1. Proses Putaran Motor DC. ....	9
Gambar 2. Motor Dc Sederhana .....	10
Gambar 3. Medan Magnet Yang Membawa Arus Mengelilingi Konduktor .....	11
Gambar 4. Prinsip Kerja Motor DC. ....	11
Gambar 5. E.M.F Kembali.....	13
Gambar 6. Simbol Dan Struktur Dioda.....	14
Gambar 7. Regulator Tegangan Positif (7812) Dan Negatif (7912).....	16
Gambar 8. Catu Daya Teregulasi Tegangan Positif.....	17
Gambar 9. Prinsip Pemasangan Transistor Eksternal NPN Pada Regulator Tegangan Tetap .....	18
Gambar 10. Bentuk Fisik <i>Push Button</i> .....	18
Gambar 11. Simbol Diagram <i>Relay</i> . ....	19
Gambar 12. IC ATmega8. ....	20
Gambar 13. Konfigurasi Dan Fungsi Kaki Pin ATmega8 .....	21
Gambar 14. Bentuk Fisik LCD LMB162AFC.....	22
Gambar 15. Contoh Bentuk Fisik Resistor.....	23
Gambar 16. Pemberian Bias Maju.....	24
(a). Transistor PNP.....	24
(b). Transistor NPN.....	24
Gambar 17. Rangkaian Transistor Sebagai Saklar.....	27
Gambar 18. Blok Diagram Sistem Rancangan .....	30
Gambar 19. Catu Daya Teregulasi Tegangan Positif.....	35
Gambar 20. Skema Rangkaian Catu Daya ( <i>Power Supply</i> ).....	36
Gambar 21. Skema Rangkaian Sistem Mikrokontroler ATmega8.....	39
Gambar 22. Skema Rangkaian LCD Monitor 16x2 Karakter.....	40
Gambar 23. Skema Rangkaian Tombol <i>Setting</i> .....	41

Gambar 24. Skema Rangkaian Saklar <i>Relay</i> .....	42
Gambar 25. Desain Boks Rangkaian Kontrol.....	44
Gambar 26. Desain Boks Motor .....	45
Gambar 27. Desain Kincir Air .....	46
Gambar 28. Desain Sirip – Sirip Kincir Air.....	47
Gambar 29. Desain Jadi Kincir Air.....	47
Gambar 30. Penyambungan As Penghubung Motor ke Kincir .....	48
Gambar 31. Desain Mekanik.....	49
Gambar 32. Diagram Alir Program Utama.....	50
Gambar 33. Diagram Alir Program <i>Setting</i> Menu. ....	51
Gambar 34. Diagram Alir Program <i>Setting</i> Jam Alat .....	52
Gambar 35. Diagram Alir Program <i>Setting</i> Alat.....	53
Gambar 36. Diagram Alir Program <i>Setting</i> Hidup Alat 1.....	54
Gambar 37. Diagram Alir Program <i>Setting</i> Hidup Alat 2.....	55
Gambar 38. Diagram Alir Program <i>Setting</i> Mati Alat 1. ....	56
Gambar 39. Diagram Alir Program <i>Setting</i> Mati Alat 2.. ....	57

## DAFTAR TABEL

	Halaman
Tabel 1. Berbagai Tipe Regulator Beserta Batasan Tegangan Masukan..	16
Tabel 2. Simbol-Simbol dalam Diagram Alir.....	28
Tabel 3. Bahan Pembuatan Rangkaian Catu Daya. ....	32
Tabel 4. Bahan Pembuatan Rangkaian Sistem Minimum. ....	33
Tabel 5. Bahan Pembuatan Rangkaian <i>LCD</i> Monitor .....	34
Tabel 6. Bahan Pembuatan Rangkaian Tombol <i>Seting</i> .....	34
Tabel 7. Bahan Pembuatan Rangkaian Saklar Relay.....	35
Tabel 8. Rencana Pengujian Trafo 15 Volt <i>DC</i> (Setelah Trafo).....	62
Tabel 9. Rencana Pengujian <i>Power Supply</i> 6 V <i>DC</i> .....	62
Tabel 10. Rencana Pengujian <i>Power Supply</i> 12 V <i>DC</i> .....	62
Tabel 11. Rencana Pengujian Saklar Relay .....	63
Tabel 12. Rencana Pengujian Fungsi Tombol .....	63
Tabel 13. Rencana Pengujian Kesesuaian Waktu Sesungguhnya Dengan Waktu Pada Alat .....	64
Tabel 14. Rencana Pengujian Kesesuaian Waktu Seting Kincir Dengan Unjuk Kerja Kincir Saat Hidup Dan Mati .....	64
Tabel 15. Hasil Pengujian Trafo 15 V <i>DC</i> (Setelah Diode) .....	66
Tabel 16. Hasil Pengujian <i>Power Supply</i> 6 V <i>DC</i> .....	67
Tabel 17. Hasil Pengujian <i>Power Supply</i> 12 V <i>DC</i> .....	68
Tabel 18. Hasil Pengujian Saklar Relay .....	70
Tabel 19. Hasil Pengujian Fungsi Tombol <i>Setting</i> .....	70
Tabel 20. Hasil Pengujian Kesesuaian Waktu Antara Waktu Sesungguhnya Dengan Waktu Pada Alat.....	71
Tabel 21. Hasil Pengujian Kesesuaian Waktu Seting Kincir Dengan Unjuk Kerja Kincir Saat Hidup Dan Mati .....	71

## DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Gambar PCB Rangkaian Sistem Minumum dan Catu Daya ....	81
Lampiran 2. Gambar PCB Rangkaian Tombol <i>Setting</i> .....	81
Lampiran 3. Gambar PCB Rangkaian Tombol <i>Reset</i> dal <i>LED</i> .....	82
Lampiran 4. Foto-Foto Proyek Akhir .....	82
Lampiran 5. Program Mikrokontroller AVR ATmega8 Dengan Bahasa C...	83



# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Ilmu pengetahuan dan teknologi mengalami perkembangan yang amat pesat pada zaman modern ini. Keadaan tersebut berimbas pada berbagai bidang yang meliputi bidang: Industri, Peternakan, Pertanian, Perikanan, dan lain-lain. Berbagai teknologi-teknologi diterapkan pada bidang tersebut untuk memperoleh hasil maksimal. Penerapan teknologi semakin marak digunakan karena mampu meningkatkan efisiensi pada proses produksi dan peningkatan kualitas hasil produksi yang tentu saja akan meningkatkan keuntungan.

Bidang perikanan merupakan salah satu bidang yang memanfaatkan perkembangan teknologi saat ini. Potensi yang besar dalam bidang perikanan baik perikanan tangkap dan perikanan budidaya menjadikan teknologi menjadi salah satu faktor penting dalam perkembangannya. Penggunaan *Global Positioning System* (GPS) pada nelayan, automatic feeder, kincir air, dan berbagai alat laboratorium menjadi bukti pentingnya teknologi dalam dunia perikanan.

Perikanan budidaya merupakan bidang perikanan yang semakin gencar dikembangkan untuk tetap menjaga hasil perikanan yang maksimum dan berkelanjutan (*Maximum Sustainable Yield*). Salah satu spesies yang populer di masyarakat adalah ikan lele. Bidang budidaya ikan lele pada dasarnya masih terkendala dengan belum dikembangkannya

sistem budidaya dengan sistem intensif terutama pada pembenihan ikan lele yang mampu meningkatkan Kuantitas serta Kualitas dari benih ikan lele yang dihasilkan.

Kendala yang kerap dialami dalam pembenihan ikan lele adalah kurangnya ketersediaan oksigen pada malam hari yaitu saat fitoplankton tidak berfotosintesis sehingga ketersediaan oksigen sangat terbatas dan keadaan ini biasa disebut titik kritis. Titik kritis terjadi antara pukul 24.00 – 05.00 dimana kandungan oksigen di perairan hanya berkisar 4 ppm. Padahal ikan lele mampu tumbuh dengan baik pada ketersediaan oksigen >7 ppm dan kondisi oksigen 4 ppm tersebut menjadikan pertumbuhan ikan lele menjadi kurang maksimal.

Melihat dari keadaan tersebut perlu adanya alat yang dapat menjaga ketersediaan oksigen terutama pada saat titik kritis yaitu kincir air dengan timer. Penggunaan alat ini diharap akan mampu menyuplai oksigen dan sistem kerjanya yang otomatis akan mampu meningkatkan efisiensi kerja dan peningkatan hasil yang lebih baik.

**B. Identifikasi Masalah**

1. Rancangan perangkat keras (*hardware*) elektronik berupa rangkaian alat kincir air otomatis berbasis mikrokontroller.
2. Rancangan dan pengaturan timer pada mikrokontroler untuk menyetting waktu pengoperasian sesuai kebutuhan secara otomatis.
3. Besar kincir air yang hanya untuk memasok kolam ukuran 3m x 5m.

**C. Batasan Masalah**

Berdasarkan pada pokok permasalahan yang telah diuraikan pada bagian identifikasi masalah diatas, maka pembuatan proyek akhir ini hanya dibatasi pada motor DC yang digunakan untuk menggerakkan kincir air dan pembuatan kincir air yang digunakan untuk ukuran kolam maksimal 3m x 5m.

**D. Rumusan Masalah**

1. Bagaimana rancang bangun kincir air dengan timer berbasis mikrokontroller dapat bekerja dengan baik?
2. Bagaimana unjuk kerja dari rangkaian kincir air dengan timer berbasis mikrokontroller?

### **E. Tujuan**

Tujuan dari tugas akhir ini adalah:

1. Membuat kincir air dengan timer berbasis mikrokontroler sehingga dapat menyuplai oksigen pada saat titik kritis yang selama ini menjadi kendala pada petani pembenihan ikan lele.
2. Mengetahui unjuk kerja kincir air dengan timer berbasis mikrokontroler.

### **F. Manfaat**

Pembuatan tugas akhir ini diharapkan dapat memberikan manfaat bagi semua pihak, yaitu:

1. Bagi mahasiswa:
  - a. Mahasiswa dapat mengasah kemampuan dalam menciptakan inovasi tidak hanya di bidang industri tetapi juga di bidang lain seperti perikanan.
  - b. Mahasiswa dapat mengaplikasikan ilmu yang telah diperoleh dalam perkuliahan.
2. Bagi Masyarakat:
  - a. Menambah khasanah rangkaian pengendali elektronik yang dapat diaplikasikan dalam bidang perikanan khususnya pembenihan ikan lele.
  - b. Dengan adanya peralatan ini akan lebih mempermudah penyuplaian oksigen pada perairan sehingga akan meningkatkan kualitas dan kuantitas benih ikan lele.

3. Bagi Institusi :

- a. Dapat dijadikan sebagai bahan pengajaran di kampus.
- b. Dapat dijadikan sebagai referensi dalam pengembangan kreatifitas mahasiswa.

**G. Keaslian Gagasan**

Pembuatan proyek akhir dengan judul “Kincir Air Alternatif dengan Timer sebagai penyuplai Kandungan Oksigen (*dissolved oxygen*) pada Pembenihan Ikan Lele berbasis mikrokontroler“ merupakan duplikasi dan pengembangan dari alat yang telah ada dan terinspirasi dari seorang petani ikan lele yaitu Yahya Hamitisna S.Pi. yang pernah bekerja juga di Balai Benih Ikan di daerah solo. Dari file-file yang diperlihatkan kepada saya terdapat gambar-gambar alat yang memberi ide saya untuk membuat proyek akhir ini. Ide yang saya dapat yaitu untuk diaplikasikan pada kolam-kolam petani ikan yang relatif kecil ukuran kolamnya, kisaran 3m x 5m. Pengembangan karya tersebut berupa rancangan mekanik, dan actuator yang berbasis electrical, yang berbentuk kincir air dengan timer untuk pengaturan waktu on dan off alat yang bisa diseting manual berbasis mikrokontroler.

## **BAB II**

### **PENDEKATAN PEMECAHAN MASALAH**

Pengaplikasian alat ini berawal dari sebuah tambak udang yang salah satunya berfungsi sebagai penyuplai kandungan oksigen selain untuk mengumpulkan sampah – sampah yang ada pada air tambak dengan sistem putaran kincir air yang membuat air menjadi bergelombang sehingga sampah – sampah dapat tersingkirkan akibat gelombang air yang dihasilkan oleh putaran kincir air. Dalam perairan tambak, alat ini menggunakan mesin diesel yang dimodifikasi dengan gardan sebagai sambungan kincir air.

Pengoperasian alat ini masih secara manual saat menghidupkan atau mematikannya, dan letaknya yang berada pada tengah tambak sehingga saat menghidupkan atau mematikannya harus menggunakan perahu sampan. Melihat keadaan tersebut dirasa alat ini kurang efisien dalam pengoperasian. Meskipun demikian alat ini sudah banyak diaplikasikan di beberapa tempat yang khususnya dalam bidang perikanan yaitu pada waduk gajah mungkur yang digunakan untuk pemeliharaan udang.

Fungsi kincir air didalam operasional kolam ikan

- a. Sebagai penyuplai oksigen didalam perairan kolam. Didalam suatu ekosistem perairan kolam kebutuhan oksigen telah disuplai oleh phytoplankton, tapi kebutuhan oksigen tersebut tidak akan mencukupi bagi biota dan proses-proses yang terjadi didalamnya. Oksigen didalam perairan tambak diperlukan tidak hanya dalam proses respirasi

(pernapasan) tapi juga dibutuhkan dalam proses-proses fisika, kimia dan biologi yang terjadi didalam perairan tersebut.

Keberadaan kincir air didalam kolam diharapkan dapat membantu dan mengantisipasi terjadinya kekurangan oksigen yang dapat terjadi pada saat tertentu didalam perairan tersebut.

- b. Membantu dalam proses pencampuran karakteristik antara perairan tambak lapisan atas, dan bawah. Sebagai suatu perairan yang statis dan memiliki ketinggian tertentu, maka suatu perairan kolam jika dalam kondisi diam akan memiliki karakteristik yang berbeda-beda antara lapisan atas dan lapisan bawah. Perbedaan karakteristik perairan tersebut, jika tidak segera diantisipasi dapat membahayakan kehidupan ikan yang ada didalamnya.

Pengoperasian kincir diharapkan dapat membantu mengantisipasi terjadinya perbedaan yang cukup menyolok antar lapisan tambak, sehingga kualitas air yang dihasilkan relatif sama antar lapisan kolam.

- c. Membantu dalam proses pemupukan air. Kegiatan pemupukan air dilakukan sebagai upaya pembentukan kualitas air yang terkait dengan kecerahan air dan warna air kolam dengan cara menstimulasi pertumbuhan phytoplankton ke arah yang lebih stabil.

Pengoperasian kincir diharapkan dapat membantu proses penyebaran pupuk secara merata didalam perairan kolam sekaligus menstimulasi pertumbuhan plankton melalui oksigen yang dihasilkannya.

- d. Membantu dalam mengarahkan kotoran dasar tambak ke arah sentral pembuangan, sehingga memudahkan dalam proses pembersihan dasar tambak. (Sumber: Agus dkk, 2008)

Dalam rancang bangun kincir air ini memerlukan beberapa komponen elektronik yang berfungsi sebagai kontrol kincir air dengan timer berbasis mikrokontroler yang diaplikasikan pada kolam pembenihan ikan lele. Komponen – komponen yang diperlukan yaitu:

#### **A. Motor DC**

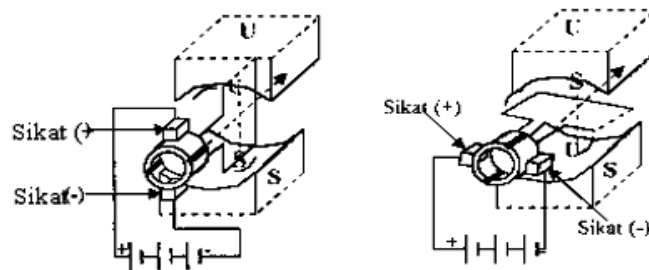
Motor DC adalah peralatan elektromekanik dasar yang berfungsi untuk mengubah tenaga listrik menjadi tenaga mekanik yang desain awalnya diperkenalkan oleh Michael Faraday. (Endra Pitowarno, 2006: 76).

Motor DC berputar dikarenakan terdapat interaksi antara medan magnet permanen dengan gaya yang bekerja pada lilitan (kumparan) karena arus yang mengalir pada lilitan tersebut. Motor berputar karena antara magnet permanen dengan gaya yang timbul pada kumparan akan saling tarik-menarik dan saling tolak-menolak pada masing-masing kutub.

Sesuai dengan namanya, motor DC diberi daya dengan tegangan DC (*Direct Current* atau arus searah). Dengan demikian putaran motor DC akan berbalik arah jika polaritas tegangan yang diberikan juga diubah. Motor DC mempunyai tegangan kerja yang bervariasi. Motor DC dipakai pada sistem robot, putarannya biasanya terlalu kencang. Untuk itu dipakai susunan *gear-gear* untuk mengurangi kecepatan putar *shaft* motor dan juga agar torsi



meningkat. Gambar 1 menunjukkan proses putaran motor DC. Poros motor DC akan berputar dari kutub positif ke kutub negative sumber. Perhatikan Gambar 1.

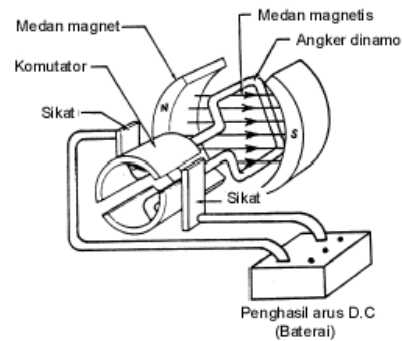


Gambar 1. Proses Putaran Motor DC

Motor listrik merupakan perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk memutar kincir yang digunakan untuk mengaduk air nantinya.

Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Jika terjadi putaran pada kumparan jangkar dalam pada medan magnet, maka akan timbul tegangan (GGL) yang berubah – ubah arah pada setiap setengah putaran, sehingga merupakan tegangan bolak – balik. Prinsip kerja dari arus searah adalah membalik fasa tegangan gelombang yang mempunyai nilai positif menggunakan komutator. Dengan demikian arus yang berbalik arah dengan kumparan jangkar yang berputar dalam medan magnet. Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen.

Gambar 2 menunjukkan motor dc sederhana dan juga bagian – bagian dari motor DC.

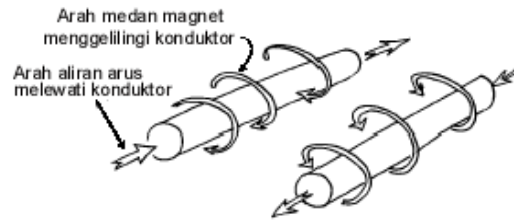


Gambar 2. Motor DC Sederhana.

Catu tegangan dc dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumputan satu lilitan pada Gambar 2 disebut angker dinamo. Angker dinamo adalah sebutan untuk komponen yang berputar di antara medan magnet. (Hamzah Berahim: 1991)

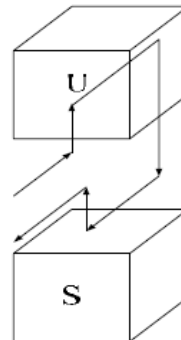
### 1) Prinsip dasar dan cara kerja motor DC

Jika arus lewat pada suatu konduktor, timbul medan magnet di sekitar konduktor. Arah medan magnet ditentukan oleh arah aliran arus pada konduktor. Gambar 3 menunjukkan arah medan magnet yang membawa arus mengelilingi konduktor.



Gambar 3. Medan Magnet Yang Membawa Arus Mengelilingi Konduktor.

Pada motor dc, daerah kumparan medan yang dialiri arus listrik akan menghasilkan medan magnet yang melingkupi kumparan jangkar dengan arah tertentu. Konversi dari energi listrik menjadi energi mekanik (motor) maupun sebaliknya berlangsung melalui medan magnet, dengan demikian medan magnet disini selain berfungsi sebagai tempat untuk menyimpan energi, sekaligus sebagai tempat berlangsungnya proses perubahan energi, daerah tersebut dapat dilihat pada Gambar 4 di bawah ini :



Gambar 4. Prinsip Kerja Motor Dc

Agar proses perubahan energi mekanik dapat berlangsung secara sempurna, maka tegangan sumber harus lebih besar daripada tegangan gerak yang disebabkan reaksi lawan. Dengan memberi arus pada

kumparan jangkar yang dilindungi oleh medan maka menimbulkan perputaran pada motor.

Dalam memahami sebuah motor, penting untuk mengerti apa yang dimaksud dengan beban motor. Beban dalam hal ini mengacu kepada keluaran tenaga putar / torque sesuai dengan kecepatan yang diperlukan. Beban umumnya dapat dikategorikan ke dalam tiga kelompok :

- **Beban torque konstan** adalah beban dengan permintaan keluaran energinya bervariasi pada kecepatan operasinya namun *torquencya* konstan. Contoh beban dengan *torque* konstan adalah *corveyors*, *rotary kilns*, dan pompa *displacement* konstan.
- **Beban dengan variabel *torque*** adalah beban dengan *torque* yang bervariasi dengan kecepatan operasi. Contoh beban dengan variabel *torque* adalah pompa sentrifugal dan *fan* (*torque* bervariasi sebagai kuadrat kecepatan).
- **Beban dengan energi konstan** adalah beban dengan permintaan *torque* yang berubah dan berbanding terbalik dengan kecepatan. Contoh untuk beban dengan daya konstan adalah pada peralatan – peralatan mesin.

## 2) Prinsip Arah Putaran Motor

Untuk menentukan arah putaran motor digunakan kaedah *Flamming* tangan kiri. Kutub-kutub magnet akan menghasilkan medan magnet dengan arah dari kutub utara ke kutub selatan. Jika medan magnet memotong sebuah kawat penghantar yang dialiri arus searah dengan empat jari, maka akan

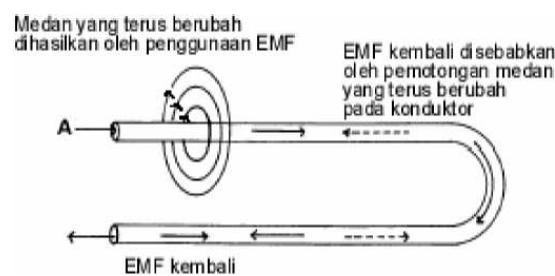
timbul gerak searah ibu jari. Gaya ini disebut gaya *Lorentz*, yang besarnya sama dengan  $F$ .

Pada prinsip kerja motor aliran arus di dalam penghantar yang berada di dalam pengaruh medan magnet akan menghasilkan gerakan. Besarnya gaya pada penghantar akan bertambah besar jika arus yang melalui penghantar bertambah besar.

### 3) Electromotive Force (EMF) / Gaya Gerak Listrik

EMF induksi biasanya disebut EMF Counter. atau EMF kembali. EMF kembali artinya adalah EMF tersebut ditimbulkan oleh angker dinamo yang melawan tegangan yang diberikan padanya.

Teori dasarnya adalah jika sebuah konduktor listrik memotong garis medan magnet maka timbul ggl pada konduktor. Gambar 5 menjelaskan tentang EMF induksi atau yang biasa disebut dengan EMF kembali.



Gambar 5. E.M.F. Kembali

EMF induksi terjadi pada motor listrik, generator serta rangkaian listrik dengan arah berlawanan terhadap gaya yang menimbulkannya.

HF. Emil Lenz mencatat pada tahun 1834 bahwa arus induksi selalu berlawanan arah dengan gerakan atau perubahan yang menyebabkannya. Hal ini disebut sebagai Hukum *Lenz*.

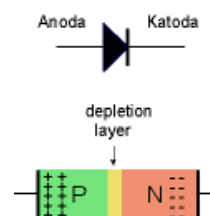
Timbulnya EMF tergantung pada beberapa hal berikut ini:

- kekuatan garis fluks magnet
- jumlah lilitan konduktor
- sudut perpotongan fluks magnet dengan konduktor
- kecepatan konduktor memotong garis fluks magnet
- Tidak adanya arus induksi yang terjadi jika angker dinamo diam.

(Hamzah Berahim: 1991)

## B. Dioda

Dalam berbagai rangkaian elektronika komponen semikonduktor dioda sering kita jumpai jenis dan tipe yang berbeda-beda tergantung dari model dan tujuan penggunaan rangkain tersebut dibuat. Dioda mempunyai dua buah elektroda yaitu anoda dan katoda. Anoda digunakan untuk polaritas positif dan katoda untuk polaritas negatif. Di dalam dioda terdapat *junction* (pertemuan) di mana daerah semikonduktor tipe-P dan semi konduktor tipe-N bertemu. Lebih jelasnya perhatikan Gambar 6. Gambar 6 menunjukkan symbol dan struktur diode.



Gambar 6. Simbol dan Struktur Dioda

(Alam, 2012 : 12)

Dioda semikonduktor hanya dapat melewatkan arus pada satu arah saja, yaitu pada saat dioda memperoleh catu arah maju (*forward bias*). Sedangkan bila dioda diberi catu arah terbalik (*reverse bias*) maka dioda tidak bekerja dan pada kondisi ini dioda mempunyai tahanan yang tinggi sehingga arus sulit mengalir. Dari kondisi tersebut maka dioda hanya digunakan pada beberapa pemakaian saja antara lain sebagai penyearah gelombang (*rectifier*), disamping kegunaan-kegunaan lainnya misalnya sebagai *klipper*, pengganda tegangan, dan lain-lain

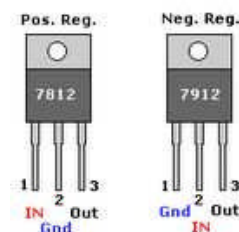
### C. Catu Daya

Perangkat elektronika mestinya dicatu oleh suplai arus searah *DC* yang stabil agar dapat bekerja dengan baik. Baterai atau *accu* adalah sumber catu daya *DC* yang paling baik. Namun untuk aplikasi yang membutuhkan catu daya lebih besar, sumber dari baterai tidak cukup. Sumber catu daya yang besar adalah sumber bolak-balik atau *Alternating Current (AC)* dari pembangkit tenaga listrik. Untuk itu diperlukan suatu perangkat catu daya yang dapat mengubah arus *AC* menjadi *DC*.

Suplai daya atau tegangan catu suatu rangkaian elektronik yang berubah-ubah besarnya dapat menyebabkan pengaruh yang sifatnya merusak fungsi kerja rangkaian elektronik yang dicatunya. Catu daya yang stabil dan dapat diatur sering disebut dengan *regulated power supply*. Catu daya ini menggunakan komponen aktif sehingga harganya cukup mahal. Maka dari itu, saat ini banyak digunakan catu daya dalam bentuk IC yaitu IC regulator tegangan. IC regulator tegangan secara garis besar dapat dibagi menjadi dua

yakni regulator tegangan tetap (3 kaki) dan regulator tegangan yang dapat diatur (3 kaki atau lebih).

IC regulator tegangan tetap yang sekarang populer adalah keluarga 78xx untuk tegangan positif dan seri 79xx untuk tegangan negatif. Bentuk IC dan susunan kakinya adalah seperti terlihat pada Gambar 7. Besarnya tegangan keluaran IC seri 78xx dan 79xx ini dinyatakan dengan dua angka terakhir pada serinya. Contoh IC 7812 adalah regulator tegangan positif dengan tegangan keluaran 12 volt, IC 7912 adalah regulator tegangan negatif dengan tegang keluaran -12 volt.



Gambar 7. Regulator Tegangan Positif (7812) dan Negatif (7912)

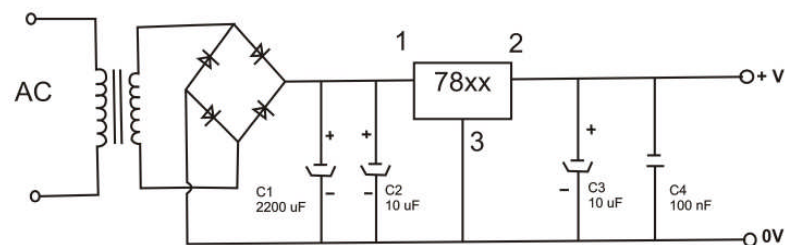
Besarnya tegangan masukan ( $V_{in}$  dalam nilai DC) pada regular seri 78xx dalam beberapa variasi tegangan keluaran dapat dilihat dalam Tabel 1 berikut:

Tabel 1. Berbagai Tipe Regulator Beserta Batasan Tegangan Masukan

Tipe Rugulator	$V_o$	$V_{in \text{ min}}$	$V_{in \text{ maks}}$
7805	5 V	7 V	20 V
7806	6 V	8 V	21 V
7808	8 V	10,5 V	25 V
7810	10 V	12,5 V	25 V
7812	12 V	14,5 V	27 V
7815	15 V	17,5 V	30 V
7818	18 V	21 V	33 V
7824	24 V	27 V	38 V



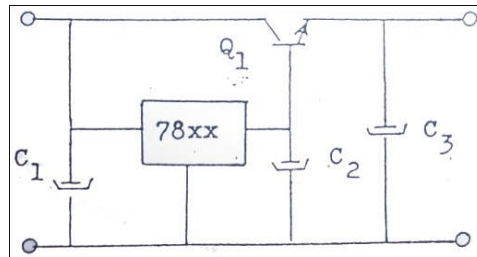
Batasan nilai tegangan masukan IC regulator yang terdapat dalam tabel adalah nilai DC, dan bukan tegangan sekunder trafo. Dengan demikian dapat ditarik kesimpulan bahwa catu daya teregulasi adalah catu daya yang dapat menghasilkan tegangan keluaran yang nilai/harga tegangannya senantiasa selalu tetap setiap saat sesuai dengan yang diharapkan. (Sunomo, 1996: 82). Perhatikan Gambar 8, gambar 8 menunjukkan rangkaian catu daya teregulasi tegangan positif.



Gambar 8. Catu Daya Teregulasi Tegangan Positif

Pemakaian *heatshink* (aluminium pendingin) dianjurkan jika komponen ini dipakai untuk mencatu arus yang besar. Di dalam *datasheet*, komponen IC regulator tegangan maksimal bisa dilewati arus mencapai 1 Ampere. Kemampuan memberikan catu daya dari IC regulator tegangan dapat ditingkatkan kapasitasnya dengan menambahkan transistor luar atau eksternal, baik transistor NPN maupun PNP. Dengan penambahan transistor luar, maka sebagian besar dari arus akan dilewatkan pada transistor luar ini, sehingga IC regulator tegangan hanya berfungsi sebagai pengontrol tegangan saja. Transistor yang sering digunakan adalah transistor 2N 3055. Perhatikan

Gambar 9. Gambar 9 menunjukkan prinsip pemasangan transistor eksternal NPN pada regulator tegangan tetap.

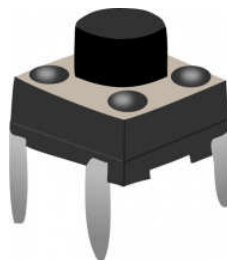


Gambar 9. Prinsip Pemasangan Transistor Eksternal NPN pada Regulator Tegangan Tetap

#### D. Push Button

*Push button* merupakan saklar yang dioperasikan secara manual. *Push button* ini berfungsi untuk memutuskan atau menghubungkan aliran listrik. Ada dua macam *push button*, yaitu *push button NO (Normaly Open)* dan *push button NC (Normaly Close)*.

*Push button NO* menghubungkan rangkaian ketika ditekan dan kembali ke posisi terbuka ketika dilepas. Sebaliknya *push button NC* membuka rangkaian ketika *push button* ditekan dan kembali pada posisi menutup ketika *push button* dilepas. Bentuk fisik dari *push button* bisa dilihat pada Gambar 10.

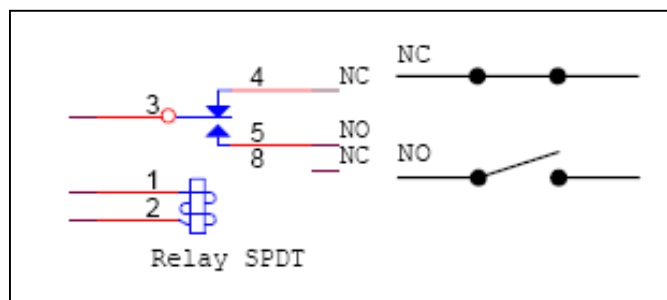


Gambar 10. Bentuk Fisik *Push Button*

(Alam, 2012: 20)

## E. Relay

*Relay* adalah sebuah piranti elektromekanik yang dioperasikan berdasarkan variasi masukan, untuk mengontrol piranti-piranti lain yang dihubungkan pada keluaran *relay*. *Relay* berfungsi untuk memutuskan atau mengalirkan arus listrik yang dikontrol dengan memberikan tegangan suplai pada koilnya. Ada dua jenis *relay* berdasarkan tegangan untuk menggerakkan koilnya, yaitu *relay DC* dan *relay AC*. Pada rangkaian ini menggunakan *relay DC* dengan tegangan 5 volt. Untuk simbol diagram *relay* dapat diperhatikan pada Gambar 11.

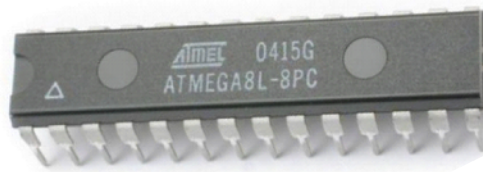


Gambar 11. Simbol Diagram *Relay*

Kontak-kontak ini dapat digunakan untuk mengontrol arus yang lebih besar dalam rangkaian. Fungsi utama *relay* adalah untuk mengontrol arus yang lebih besar dalam rangkaian dengan arus kecil yang melewati koil *relay*. Pada simbol Gambar 11 terdiri atas sebuah kumparan dan dua set kontak, satu diantaranya terbuka (*NO*), dan lainnya tertutup (*NC*). Sewaktu ada tegangan suplai pada koil *relay*, maka kontak *NO* akan terhubung dan kontak *NC* akan terbuka. Sebaliknya saat tidak ada suplai pada koil *relay* maka kontak *NO* kembali terbuka dan kontak *NC* kembali terhubung.

## F. Mikrokontroler ATmega 8

Mikrokontroler ATmega8 merupakan bagian utama dari sistem kontrol, mikrokontroler ini merupakan jenis mikrokontroler jenis AVR. Mikrokontroler jenis ini dipilih karena mikrokontroler ATmega8 memiliki 28 port masukan dan keluaran atau biasa disebut dengan port IO yang dibagi menjadi port-port B, C, dan D yang dapat difungsikan sebagai masukan dan sebagai keluaran sistem. Proses pengisian (*downloading*) program yang mudah karena memiliki fasilitas *in-system programming* yang sudah terdapat di dalam ATmega8. Lima pin, MOSI, MISO, SCK, Reset, dan Ground digunakan untuk memprogram ATmega8 ini. Bentuk fisik dari IC ATmega8 dapat dilihat pada Gambar 12.



Gambar 12. IC ATmega8

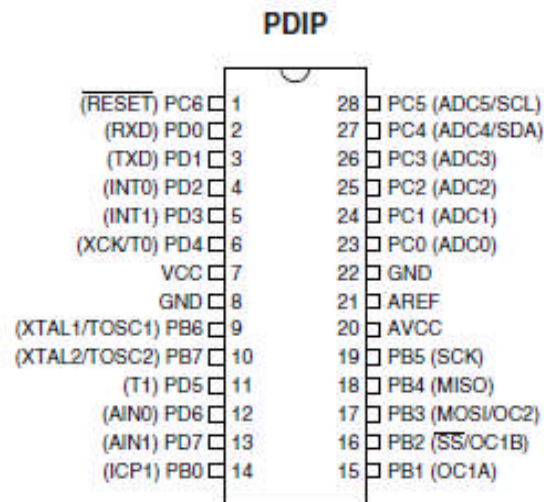
Mikrokontroler ATmega8 dapat bekerja apabila mendapat tegangan masukan sebesar 5 volt dengan batas toleransi tegangan sebesar 5,4 volt. Apabila tegangan masukan melebihi batas toleransi maka ATmega8 akan rusak dan tidak dapat digunakan kembali. Maka dari itu digunakan sistem catu daya teregulasi dengan penambahan transistor 2N 3055 sebagai penyetabil tegangan agar batas tegangan tidak melebihi dari batas toleransi. Proses pengisian pada mikrokontroler ATmega8 dapat mencapai seribu kali proses *downloading*.

## 1. Spesifikasi dari Mikrokontroler ATmega8

Setiap mikrokontroler memiliki jenis dan spesifikasi masing-masing tergantung dari kegunaan dan kebutuhan dari mikrokontroler yang akan digunakan. ATmega8 memiliki spesifikasi antara lain; 1 Kb internal SRAM, 8 Kb *flash memory*, 512 bytes EEPROM, 23 jalur *Input-Output*, 8 bit *timer/counter*, 16 bit *timer/counter*, 8-, 9-, 10- bit PWM, *On-chip Analog comparator*, Full duplex UART, SPI *serial interface for in-system programming* dan *internal power reset*.

## 2. Konfigurasi dan Fungsi Kaki Pin ATmega8

Mikrokontroler ATmega8 memiliki kaki pin sebanyak 28 buah yang terdiri dari tiga buah port, port B, port C, dan port D, dan beberapa pin lain yang memiliki fungsi dan kegunaannya masing-masing. Gambar 13 menunjukkan fungsi dan jenis kaki pin dari mikrokontroler ATmega8.

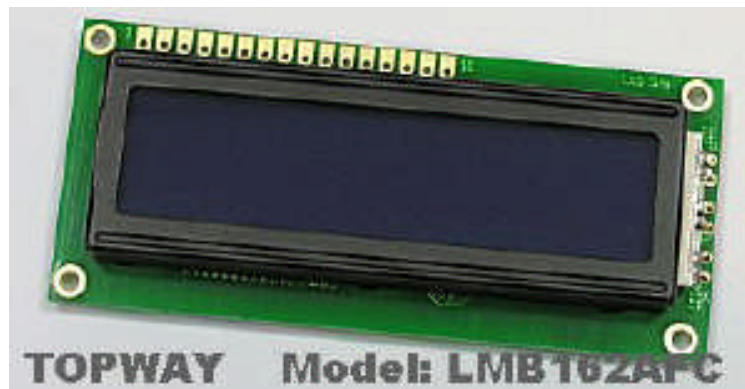


Gambar 13. Konfigurasi dan Fungsi Kaki Pin ATmega8

(Alphan, 2012: 32)

### G. *Liquid Crystal Display (LCD)*

*LCD* adalah suatu *display* dari bahan cairan kristal yang pengoperasiannya menggunakan sistem matriks. *LCD* banyak digunakan sebagai *display* pada alat-alat elektronika seperti kalkulator, multimeter digital, jam digital, dan sebagainya. *LCD* dapat dengan mudah dihubungkan dengan mikrokontroller AVR ATmega8. Modul *LCD* LMB162AFC yang merupakan *LCD* dengan *display* dua baris dengan masing-masing baris sebanyak 16 kolom. Modul *LCD* LMB162AFC dapat diakses 4 *bit* maupun 8 *bit interface*. Namun rutin-rutin pada mikrokontroller yang digunakan sudah dirancang untuk mengakses modul *LCD* ini secara 4 *bit interface*. *LCD* LM162AFC secara fisik dapat dilihat pada Gambar 14 berikut:

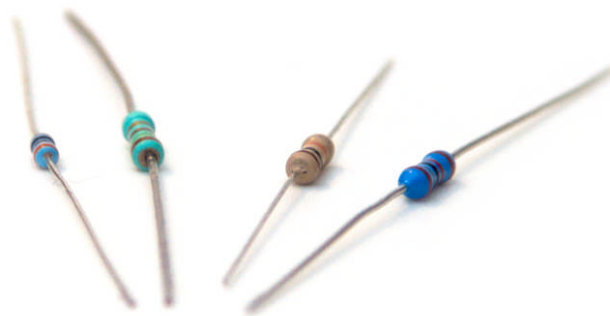


Gambar 14. Bentuk Fisik *LCD* LMB162AFC.

Rancangan *interface LCD* tidak memerlukan banyak komponen pendukung. Hanya diperlukan sebuah resistor dan sebuah variabel resistor untuk memberi tegangan kontras pada matriks *LCD*.

## H. Resistor

Resistor adalah komponen dasar elektronika yang digunakan untuk membatasi jumlah arus yang mengalir dalam suatu rangkaian. Resistor bersifat resistif dan umumnya terbuat dari bahan karbon. Satuan resistansi adalah ohm atau dilambangkan dengan simbol  $\Omega$  (omega). Tipe resistor pada umumnya berbentuk tabung dengan dua kaki tembaga di sisi kiri dan kanan. Resistor yang berbentuk tabung terdapat lingkaran gelang kode warna untuk memudahkan membaca dan mengetahui besar resistansi resistor tanpa harus mengukur dengan alat ukur. Bentuk fisik dari resistor dapat dilihat pada Gambar 15.



Gambar 15. Contoh Bentuk Fisik Resistor.

## I. Transistor

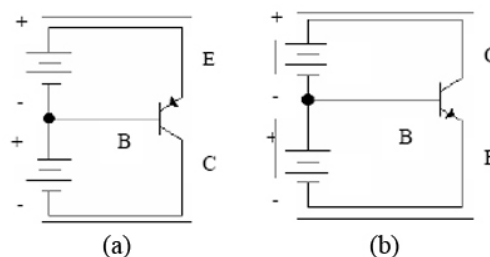
Transistor merupakan komponen aktif dengan besar arus dan tegangan atau daya keluaran dikendalikan oleh arus masukan. Transistor dibagi menjadi dua tipe yaitu sambungan *bipolar* atau disebut *Bipolar Junction*

*Transistor (BJT)* dan transistor tipe efek medan atau *Field Effect Transistor (FET)*. Transistor dari tipe sambungan *bipolar* merupakan transistor yang banyak digunakan dalam aplikasi elektronika.

Secara prinsip, transistor sambungan *bipolar* dapat dipahami sebagai sebuah sambungan (*junction*) antara dua buah dioda PN yang saling bertolak belakang. Dua buah dioda tersebut adalah dioda emitor-basis atau disebut dioda emitor dan dioda kolektor-basis atau disebut dioda kolektor.

Susunan dari dua buah dioda PN ini menentukan jenis dari transistor. Jenis transistor tersebut dapat dikelompokkan dalam dua macam, yaitu jenis NPN dan PNP. Transistor *bipolar* ini memiliki tiga buah elektroda yang masing-masing disebut dengan emitor atau *emitter (E)*, basis atau *base (B)*, dan kolektor atau *collector (C)*.

Transistor dapat bekerja jika ada bias maju (*forward bias*) yang diberikan padanya. Bias maju merupakan proses pembuatan tegangan pada bahan penyusun transistor sehingga jenis P lebih positif daripada jenis bahan N. Adanya bias maju ini memungkinkan adanya aliran elektron dari emitor ke kolektor dan arus mengalir dari kolektor ke emitor. Gambar 16 menunjukkan rangkaian pemberian bias maju pada transistor NPN dan PNP.



Gambar 16. Pemberian Bias Maju (a) Transistor PNP, (b) Transistor NPN



Transistor memiliki tiga kondisi kerja yaitu kondisi jenuh (saturasi), kondisi aktif, dan kondisi sumbat (*cut off*). Penentuan tiga macam kondisi kerja ini didasarkan pada grafik kurva kolektor dan garis beban transistor.

Transistor saklar merupakan salah satu jenis desain rangkaian transistor yang didasarkan pada dua kondisi kerja transistor, yaitu kondisi sumbat (*cut off*) dan kondisi jenuh (saturasi). Kondisi sumbat menggambarkan sebuah saklar dalam posisi terbuka. Kondisi jenuh menggambarkan sebuah saklar dalam kondisi tertutup.

Kondisi *cut off* terjadi ketika nilai arus kolektor ( $I_C$ ) sangat kecil. Sangat kecilnya arus kolektor disebabkan oleh sangat kecilnya atau tidak ada arus yang mengalir pada basis ( $I_B$ ). Saat demikian, nilai tegangan kolektor-emitor ( $V_{CE}$ ) berada dalam kondisi maksimal mendekati tegangan sumbernya ( $V_{CC}$ ).

$$1. \quad V_{CE} = V_{CC} - (I_C \times R_C)$$

Karena nilai  $I_C$  sangat kecil dan dapat dianggap nol, maka persamaan tersebut menjadi:

$$2. \quad V_{CE} = V_{CC}$$

Kondisi jenuh terjadi ketika arus kolektor ( $I_C$ ) berada pada posisi maksimum dengan nilai tegangan kolektor-emitor ( $V_{CE}$ ) mendekati nol.

Besarnya nilai arus kolektor ini ditentukan oleh besarnya perbandingan antara nilai tegangan catu pada kolektor dengan nilai hambatan kolektornya.

Persamaan yang ada yaitu:

$$3. \quad V_{CE} = 0$$

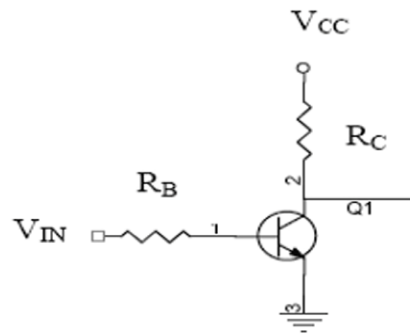
Maka:

$$4. \quad I_C = \frac{V_{CC}}{R_C}$$

Transistor akan berfungsi sebagai saklar tertutup ketika besarnya arus basis minimal sama dengan besarnya arus basis saturasi ( $I_B = I_{B \text{ saturasi}}$ ). Adapun besarnya arus basis tersebut dirumuskan dengan persamaan sebagai berikut:

$$1. \quad I_B = \frac{V_{IN} - V_{BE}}{R_B}$$

Dalam aplikasinya, transistor sebagai saklar dapat digambarkan seperti pada Gambar 17. Pada Gambar 17 transistor berfungsi sebagai saklar terbuka ketika arus basis sangat kecil atau bisa dikatakan tidak ada ( $I_B = 0$ ).



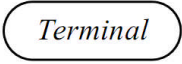
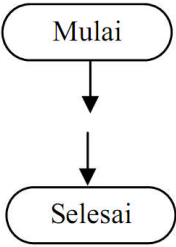
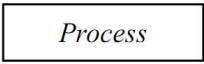
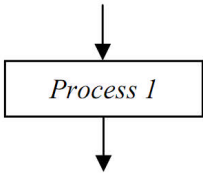

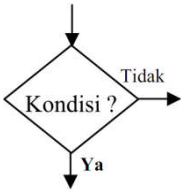
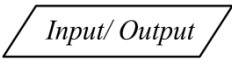
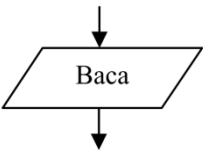
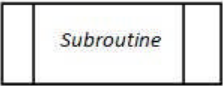
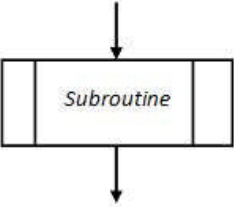
Gambar 17. Rangkaian Transistor Sebagai Saklar

#### J. Diagram Alir (*Flowchart*)

Diagram alir digunakan untuk merencanakan terlebih dahulu mengenai apa yang harus dikerjakan dari memulai sampai mengakhiri sebuah program. Dengan pembuatan struktur program yang baik, *programmer* akan mudah dalam menemukan permasalahan pada program.

Program tidak hanya dapat berjalan saja, namun juga menyertakan kemudahan akses dalam program untuk menelusuri kembali. Penelusuran kembali pada program dibutuhkan agar dapat ditemukan sebuah pengembangan dalam program tersebut. Simbol – simbol yang digunakan dalam pembuatan diagram alir dapat dilihat pada Tabel 2 berikut ini:

Tabel 2. Simbol-Simbol dalam Diagram Alir

No	Simbol	Contoh	Keterangan
1			Terminal. Simbol ini menyatakan awal dan akhir sebuah rangkaian perjalanan sebuah sistem.
2			Process. Simbol ini menyatakan adanya sebuah kegiatan pengolahan data pada langkah tersebut.
3			Decision. Simbol ini menyatakan adanya suatu alternatif (pilihan) proses.
4			Input/Output. Simbol ini digunakan untuk adanya proses pembacaan data file input, atau melakukan penulisan pada file keluaran (output).
5			Subroutine. Simbol ini menandakan akan adanya suatu percabangan rutin di mana keluar dari program utama untuk memproses isi program pada rutin percabangan terlebih dahulu. Setelah proses rutin selesai kembali ke program utama.

(Alam, 2012 : 35)

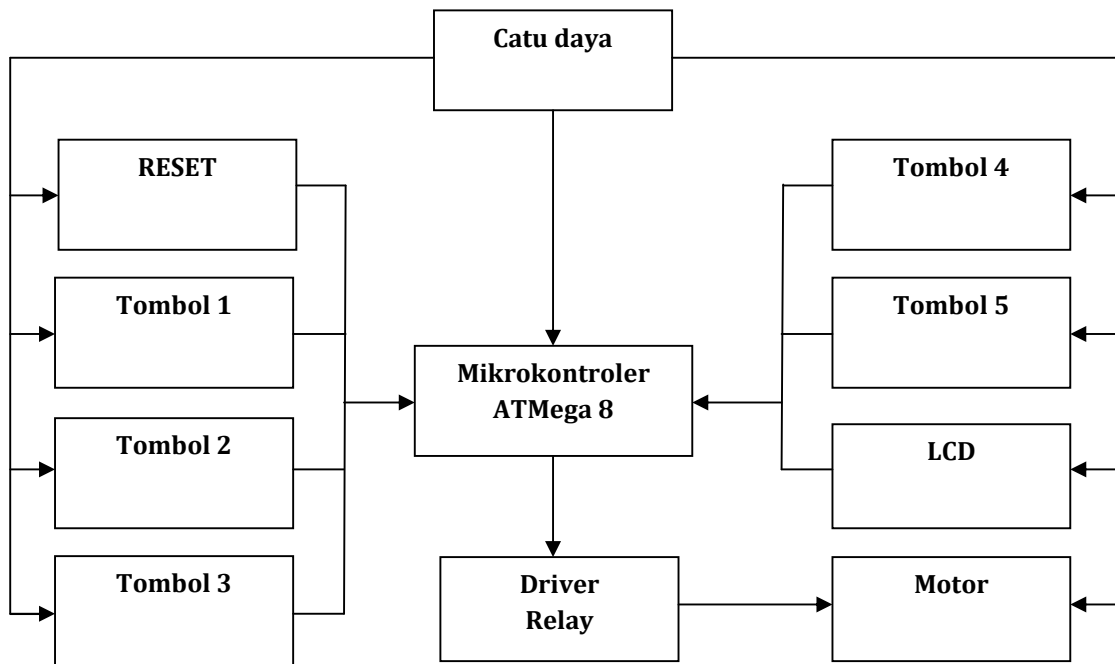
### **BAB III**

#### **KONSEP PERANCANGAN ALAT**

##### **A. Konsep Rancangan Perangkat Keras (*Hardware*)**

Perancangan pada dasarnya merupakan suatu tahapan yang sangat penting dalam pembuatan suatu alat, karena dengan menganalisa komponen yang digunakan maka alat yang dibuat dapat bekerja secara maksimal seperti yang diharapkan. Perancangan merupakan suatu hal yang dilakukan untuk mempermudah proses pembuatan alat. Perancangan terdiri dari pembuatan diagram blok dan sketsa rangkaian untuk setiap blok dengan fungsi tertentu dan spesifikasi alat yang diharapkan. Lalu setiap blok dihubungkan sehingga terbentuk sistem alat yang diharapkan.

Kincir Air Alternatif dengan Timer sebagai Penyuplai Oksigen pada Kolam Pembenihan Ikan Lele ini didesain dengan dua bagian pokok perangkat yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*). Secara garis besar perangkat keras terdiri dari mekanik dan sistem minimum mikrokontroler ATmega8 sebagai kendali utama yang berfungsi sebagai pengolah data dan pengontrol keseluruhan dari sistem, masukan atau input berupa tombol setting serta reset dan keluaran atau output berupa gerakan motor guna menggerakkan kincir air. Untuk lebih jelasnya, perhatikan skema blok diagram dapat dilihat pada Gambar 18.



Gambar18. Blok Diagram Sistem Perancangan

Bagian sistem input pada kincir air alternatif dengan timer berbasis ATmega8 ini menggunakan 6 buah tombol. Lima buah tombol berfungsi sebagai penyettingan waktu dan 1 tombol untuk reset. Lima buah tombol dalam perencanaannya nanti akan terbagi atas 1 tombol menu, 2 tombol untuk menggeser kanan dan kiri menu dan 2 tombol lagi untuk menambah atau mengurangi waktu pada saat penyettingan berlangsung. Tombol reset berfungsi untuk menghapus pengaturan jam dan bisa juga sebagai pengaman manual apabila terjadi error pada saat alat sedang bekerja. Setelah pengaturan waktu selesai, maka mikrokontroler akan memproses data inputan waktu baik penyalan kincir ataupun mematikan kincir yang kemudian diteruskan ke relay untuk menghidupkan ataupun mematikan kincir air. Penyettingan waktu

nantinya akan ditampilkan pada LCD sebagai pedoman guna mengatur waktu, menghidupkan dan juga mematikan kincir air sehingga bisa diketahui tepat tidaknya kincir beroperasi sesuai jam yang ditunjukkan pada LCD.

Dalam pembuatan Kincir air alternatif dengan timer berbasis mikrokontroler ini dapat dirinci peralatan dan bahan yang dibutuhkan, antara lain sebagai berikut:

1. Rangkaian catu daya
2. Rangkaian driver relay motor DC
3. Rangkaian mikrokontroler sebagai perangkat kontrol dan pengolah data yang merupakan satu kesatuan dari prosesor, memori, *timer/counter*, unit detak dan bagian *I/O* yang dibuat dalam satu *chip* tunggal.
4. Rangkaian *LCD* Monitor sebagai penunjuk waktu yang juga sebagai penunjuk pengaturan waktu menyala dan waktu mati kincir air secara otomatis.
5. Rangkaian tombol setting.

## **B. Identifikasi Alat dan Bahan yang Dibutuhkan**

Rangkaian catu daya dibuat dengan satu sumber yaitu sumber PLN sebagai catu daya utama. Berbagai bahan yang dibutuhkan untuk pembuatan rangkaian catu daya adalah seperti yang tercantum pada Tabel 3.

Tabel 3. Bahan Pembuatan Rangkaian Catu Daya.

No.	Bahan	Spesifikasi
1.	<i>Transformator Step Down</i>	Non CT, 3 A
2.	Dioda	1N4002
3.	Capasitor Elektrolit	2200 uF, 10 uF
4.	IC Regulator Tegangan	7806, 7812
5.	Transistor	2N 3055
6.	<i>Heatshink</i>	Untuk Transistor 2N
7.	<i>LED</i>	3mm
8.	Resistor	0.25 W, 4k7 k $\Omega$
9.	Terminal Block	2 pin
10.	Saklar <i>Power</i>	6A 250V
11.	<i>Fuse Housing</i> dan <i>Fuse</i>	1A 250V
12.	<i>Specer</i>	2 cm

Rangkaian mikrokontroler sebagai perangkat kontrol dan pengolah data adalah sebuah rangkaian sistem minimum mikrokontroler AVR ATmega8. Berbagai bahan yang diperlukan untuk pembuatan rangkaian sistem minimum adalah seperti yang tercantum pada Tabel 4.



Tabel 4. Bahan Pembuatan Rangkaian Sistem Minimum

No.	Bahan	Spesifikasi
1.	IC Atmega8	DIP 28 pin
2.	<i>Crystal</i>	11.059200 MHz
3.	Kapasitor Keramik	22 pF
4.	Kapasitor Mylar	10 uF
5	Resistor	0.25 W, 10 k $\Omega$
6	Soket <i>IDC</i>	2x5 pin
7	Dioda	1N4002
8	Soket IC	DIL 28 pin
9	White house	8 Pin
10.	White house	3 Pin
11	White house	2 Pin
12	Kabel Flat	Isi 10 kabel

Rangkaian *LCD* monitor digunakan untuk menampilkan waktu dan data seting (jam, menit dan detik) saat kincir menyala atau mati. Berbagai bahan yang diperlukan untuk pembuatan rangkaian *LCD* monitor adalah seperti yang tercantum pada Tabel 5.

Tabel 5. Bahan Pembuatan Rangkaian *lcd* Monitor.

No.	Bahan	Spesifikasi
1.	<i>LCD</i>	16x2 karakter
2.	Resistor Variabel	10 k $\Omega$
3.	Resistor	1 k $\Omega$
4.	White housing	2 pin
5.	Pin Deret Bengkok L	2x40 pin
6.	Soket IDC	5x2 pin
7.	Kabel Flat	Isi 10 kabel

Rangkaian tombol seting digunakan untuk memasukkan data (jam, menit dan detik) saat menyalakan dan mematikan kincir air. Berbagai bahan yang diperlukan untuk pembuatan rangkaian tombol seting adalah seperti yang tercantum pada Tabel 6.

Tabel 6. Bahan Pembuatan Rangkaian Tombol Seting.

No.	Bahan	Spesifikasi
1.	<i>Push Button</i>	Medium
2.	Resistor Deret	6x1 k $\Omega$
3.	White Housing	7 pin
4.	Kabel Flat	Isi 10 kabel

Rangkaian saklar *relay* berfungsi untuk mengendalikan (ON/OFF) motor motor DC yang digunakan untuk menjalankan kincir air. Berbagai bahan yang diperlukan untuk pembuatan rangkaian saklar relay adalah seperti yang tercantum pada Tabel 7.

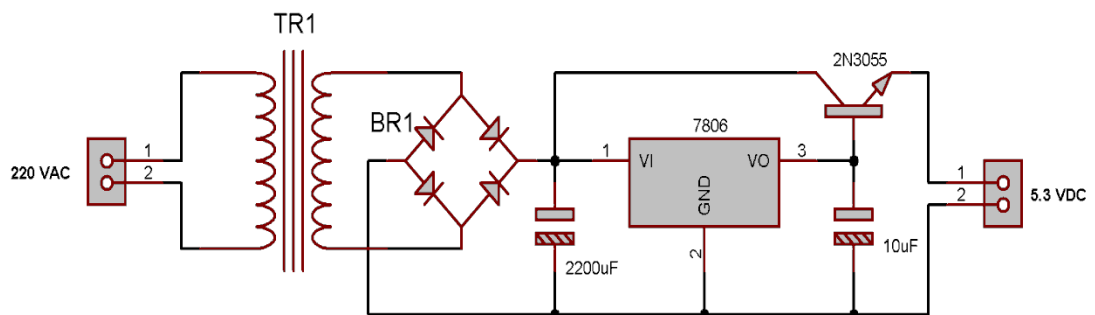
Tabel 7. Bahan Pembuatan Rangkaian Saklar Relay

No.	Bahan	Spesifikasi
1.	Transistor	BC547
2.	<i>Relay</i>	5Volt, 5 pin, Kontak 7A 250VAC
3.	Diode	In4002
4.	White house	8 Pin
5.	<i>Specer</i>	Medium
6.	Resistor	6K8 $\Omega$

### C. Perancangan Perangkat Keras (*Hardware*)

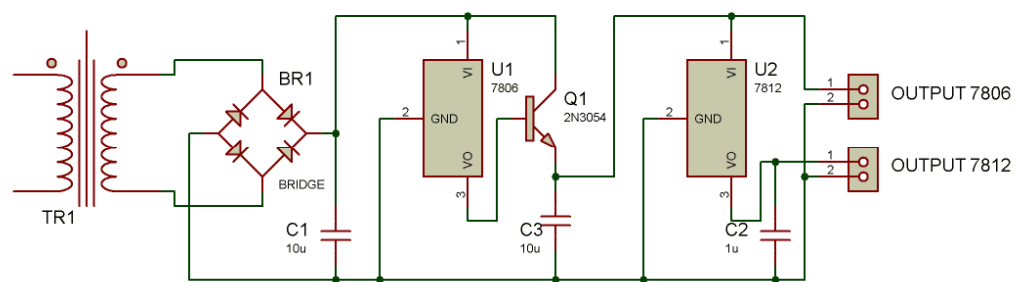
#### 1. Catu Daya (*Power Supply*)

Catu Daya (*Power Supply*) merupakan rangkaian yang menyediakan catu daya untuk setiap komponen pada rangkaian. Kincir air otomatis ini terdiri dari komponen-komponen elektronik yang membutuhkan catu daya yang stabil. Rangkaian catu daya yang akan dibuat mengacu pada rangkaian catu daya teregulasi (Sunomo, 1996). Gambar 19 merupakan rangkaian catu daya teregulasi dengan pemasangan transistor 2N 3055.



Gambar 19. Catu Daya Teregulasi Tegangan Positif.

Rangkaian tersebut harus dimodifikasi agar sesuai dengan kebutuhan. Besarnya tegangan keluaran yang dibutuhkan adalah tegangan *DC* 12 volt dan 5 Volt. Catu daya yang akan dibuat dibagi menjadi 4 bagian. Bagian pertama yaitu adaptor yang berguna untuk mengubah tegangan *AC* 220 volt menjadi tegangan *DC* 12 volt. Bagian kedua adalah regulasi tegangan keluaran menjadi 5 volt. Perhatikan Gambar 20 dimana rangkaian telah diubah menjadi dua *output* yaitu 5 volt dan 12 volt.



Gambar 20. Skema Rangkaian Catu Daya (Power Supply)

Rangkaian catu daya terdiri dari komponen *transformator stepdown*. *Transformator* ini mendapat *supply* dari tegangan jala-jala PLN, kemudian tegangan tersebut diturunkan dari 220 volt pada sisi *primer* menjadi 15 volt pada sisi *sekunder*. Tegangan *AC* 15 volt akan disearahkan menggunakan dioda jembatan. Tegangan yang keluar dari dioda *bridge* masih berdenyut, maka digunakan *electrolit condensator* 2200μF untuk menekan riak gelombang (*ripple*). Sedangkan *electrolit condensator* 10μF berfungsi untuk menjaga kestabilan tegangan keluaran

saat terjadi perubahan yang mendadak pada beban. Dari tegangan 12 volt, maka tegangan akan dimasukkan ke IC *regulator* tegangan LM7806 untuk mendapatkan tegangan *DC* 6 volt yang stabil. Setelah itu tegangan keluaran *DC* 6 volt akan dilewatkan pada transistor penguat daya 2N3055, sehingga tegangan keluaran akan stabil pada nilai 5,3 volt. Tegangan 5.3 volt telah sesuai dengan spesifikasi suplai tegangan yang tertera pada *datasheet* Mikrokontroller AVR Atmega8. Tegangan 5,3 volt ini dapat digunakan untuk mencatu Sistem Mikrokontroller, *LCD* Monitor, dan Tombol Seting. Catu daya untuk mensuplai motor kincir ini berasal dari tegangan keluaran adaptor sebesar 12 volt yang dimasukkan pada IC *regulator* tegangan LM7812.

Rangkaian Catu Daya (*Power Supply*) ini tersusun atas beberapa komponen antara lain:

a. *Transformer Step Down*

*Transformer Step Down* merupakan suatu komponen yang berfungsi untuk menurunkan tegangan *AC* 220 Volt menjadi tegangan *AC* 15 Volt.

b. *Dioda Bridge*

*Dioda Bridge* merupakan komponen yang berfungsi untuk menyearahkan gelombang *AC* menjadi gelombang *DC* yang masih kasar.

c. *Kapasitor*

Kapasitor yang digunakan adalah jenis *electrolit condensator*  $2200\mu F$  yang berfungsi sebagai perata tegangan yang telah disearahkan oleh dioda *bridge*. Sedangkan *electrolit condensator*  $10\mu F$  berfungsi sebagai penyetabil tegangan keluaran rangkaian catu daya bila terjadi perubahan beban secara tiba-tiba.

d. IC LM7806

IC LM7806 berfungsi untuk membatasi tegangan agar *output* yang keluar maksimal 6 volt *DC* yang nantinya akan digunakan untuk mencatu Sistem Mikrokontroller, *LCD* Monitor, dan Tombol *Setting*.

e. IC LM7812

IC ini berfungsi untuk membatasi tegangan *output* yang keluar maksimal 12 volt yang dipakai untuk mencatu motor DC.

f. Transistor 2N 3055

Transistor 2N 3055 berfungsi sebagai penguat daya pada catu daya teregulasi. Sebagian besar arus keluaran akan dilewatkan pada transistor 2N 3055, sehingga IC regulator tegangan hanya berfungsi sebagai pengontrol tegangan.

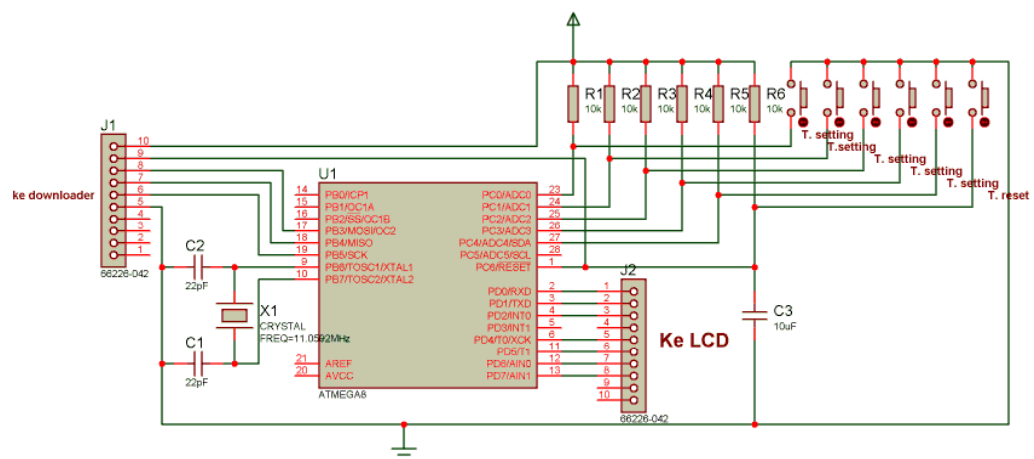
g. *Fuse*

*Fuse* berfungsi sebagai pengaman terhadap hubung singkat.

## 2. Perangkat Kontrol dan Pengolah Data

Perangkat Kontrol dan Pengolah Data pada alat ini adalah berupa Sistem Mikrokontroller AVR Atmega8. Skema dasar rangkaian sistem

minimum mengacu pada buku “Pemrograman Mikrokontroler AVR ATMEGA8 Menggunakan Bahasa C” (Heri Andrianto, 2008). Skema rangkaian sistem minimum mikrokontroller adalah seperti Gambar 22 berikut.



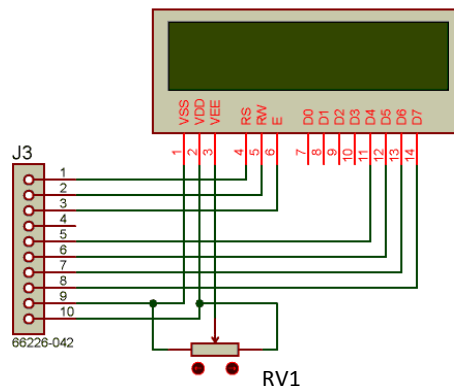
Gambar 21. Skema Rangkaian Sistem Minimum Mikrokontroller Atmega8.

Rangkaian Mikrokontroler AVR ATmega8 tersusun dari rangkaian sistem minimum yaitu IC Atmega8, *oscilator* eksternal dan *reset*. *Oscilator* eksternal berfungsi untuk menentukan kecepatan eksekusi program. *Rangkaian oscilator* eksternal terdiri dari komponen kapasitor dan *crystal* dengan nilai 11.0592 MHz. *Crystal* dengan nilai 11.0592 MHz digunakan agar didapatkan nilai yang tepat saat menggunakan fungsi *timer* dengan periode 100 ms. Tombol reset berfungsi untuk mereset mikrokontroller. Beberapa *PORT B* difungsikan sebagai pengisian program ATmega8. *PORT C* difungsikan sebagai *port* untuk tombol

*setting*. *PORT D* difungsikan sebagai *port* keluaran untuk *interface* dengan *LCD* monitor.

### 3. LCD Monitor

*LCD* monitor berfungsi untuk menampilkan seting waktu (jam, menit dan detik) saat kincir nyala ataupun mati. (Heri Andrianto, 2008). Skema rangkaian *LCD* monitor adalah seperti Gambar 23. Resistor variabel RV1 berfungsi untuk mengatur tegangan kontras tampilan karakter pada LCD.



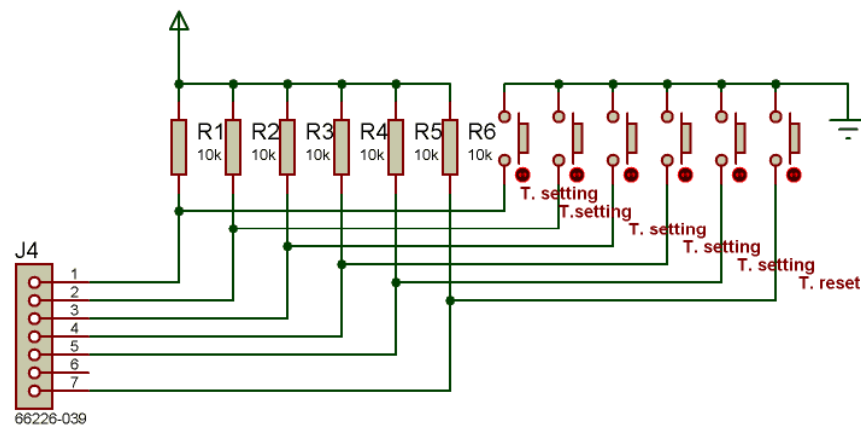
Gambar 22. Skema Rangkaian *LCD* monitor 16x2 karakter.

### 4. Tombol *Setting*

Tombol *Setting* berfungsi untuk memasukkan data seting berupa jam, menit dan detik kincir akan menyala dan mati. *Pin C bit* ke-1 sampai dengan ke – 4 difungsikan sebagai *input*. Resistor bernilai 1 k $\Omega$  berfungsi untuk membatasi arus yang masuk ke mikrokontroler. Tegangan 5 volt yang dilewatkan pada resistor sebesar 1 k $\Omega$  memberikan *input* data pada *Pin C bit* ke-1 sampai dengan ke – 4 berlogika 0. Saat tombol ditekan



maka arus akan langsung mengalir ke *ground*, sehingga *input* data pada *Pin C bit* ke-1 sampai dengan ke – 4 menjadi berlogika 1. Perhatikan Gambar 24. Gambar ini menunjukkan skema rangkaian tombol *setting*.



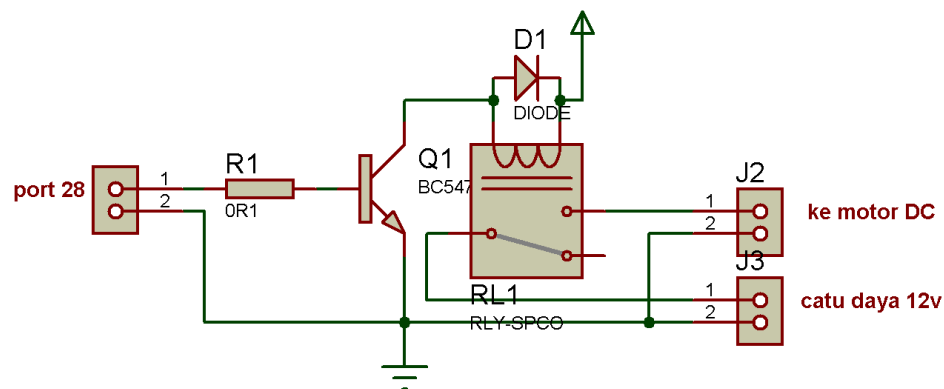
Gambar 23. Skema Rangkaian Tombol *Setting*.

## 5. Saklar Relay

Relay di sini hanya berfungsi sebagai pemutus dan penyambung tegangan sumber yang menuju ke motor penggerak kincir. Saklar relay memiliki komponen utama berupa transistor yang berfungsi sebagai saklar dan relay DC 5 Volt.

Saklar ini akan bekerja mengaktifkan relay ketika bagian basis mendapat masukan logika high atau tegangan 5 Volt oleh mikrokontroler, maka transistor akan mengalami saturasi atau tertutup, sehingga coil relay teraliri arus dan magnet yang akan menarik kontak-kontak relay. Jika basis transistor mendapatkan logika *low* atau tegangan 0 Volt, maka transistor akan berada dalam kondisi *cut-off*, sehingga *coil* hampir tidak

teraliri arus, sehingga relay *off*. Dioda *freewheeling* berfungsi untuk melindungi transistor dari tegangan balik yang terlalu besar akibat perubahan arus pada *coil*. Skema rangkaian saklar *relay* adalah seperti gambar 25.



Gambar 24. Skema Rangkaian Saklar relay

## 6. Pembuatan PCB dan Perakitan Komponen

### a. Pembuatan *PCB*

Pembuatan *PCB* dilakukan dengan menggunakan *software PCB Wizard Unlimited*. Adapun proses pembuatan *PCB* adalah sebagai berikut:

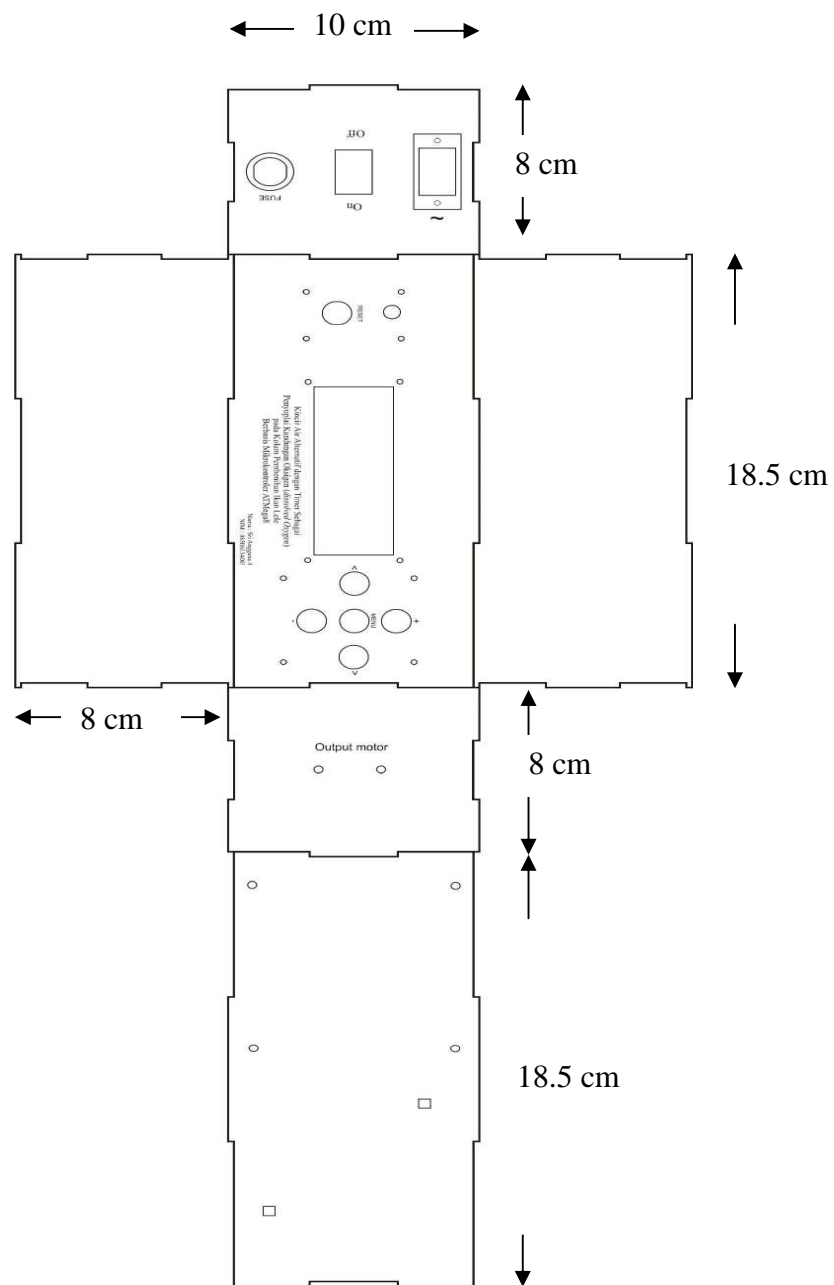
- 1) Membuat *layout PCB* menggunakan *software PCB Wizard Unlimited*. *Layout* yang sudah jadi kemudian dicetak.
- 2) *Layout* hasil cetakan kemudian di-*photo copy* menggunakan kertas *glossy*.
- 3) Sablon *layout* kertas *glossy* pada *PCB* polos menggunakan setrika listrik.

- 4) *PCB* yang sudah disablon kemudian dilarutkan menggunakan larutan *ferite clorida* ( $FeCl$ ) agar lapisan tembaga yang tidak terpakai hilang.
  - 5) Pemeriksaan jalur *PCB*.
  - 6) Lubangi *PCB* sesuai dengan pola yang tersablon menggunakan bor.
  - 7) Bersihkan *PCB* dari tinta sablon
- b. Tahap perakitan komponen

Pada tahap ini, komponen dipasang pada lubang *PCB*. Semua komponen harus dipastikan bahwa tempat ataupun posisi pemasangannya benar. Setelah selesai, maka komponen dapat disolder. Cara penyolderan harus benar, agar semua komponen terpasang dengan kuat.

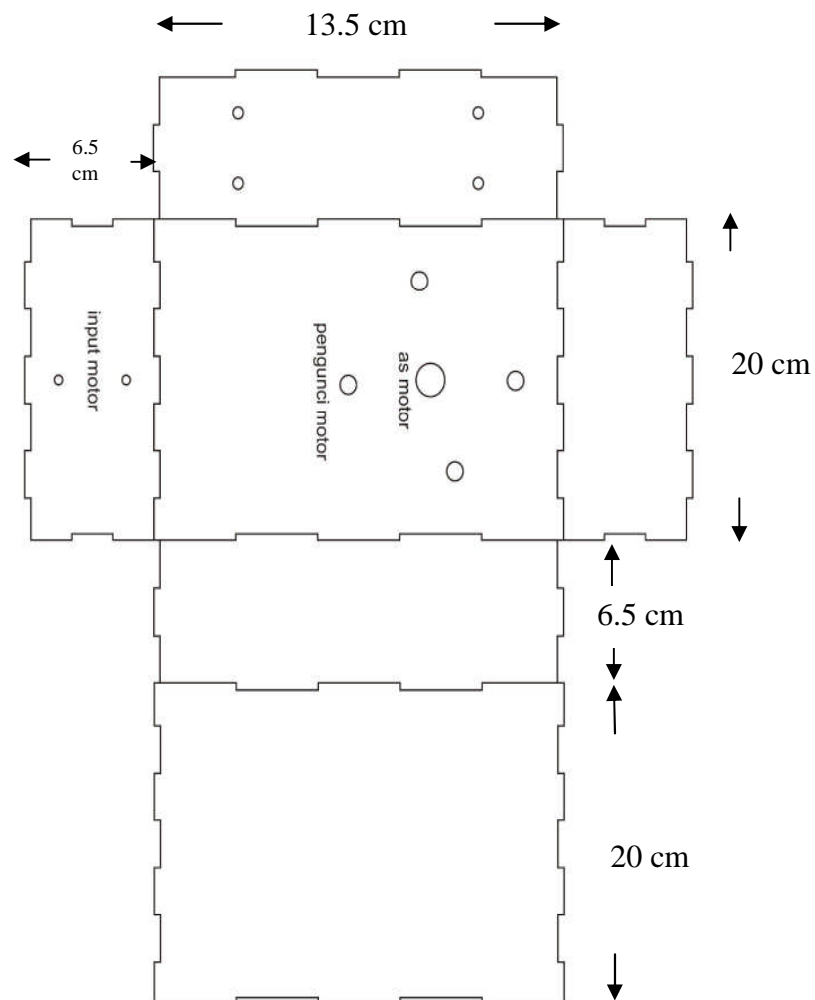
## 7. Pembuatan Boks

Pembuatan boks untuk rangkaian kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan ikan lele berbasis mikrokontoler ATmega8 menggunakan bahan akrilik transparan dengan ketebalan 2 mm. Boks dibuat berbentuk balok dengan tinggi 8 cm, lebar 10 cm, dan panjang 118.5 cm seperti Gambar 25.



Gambar 25. Desain Boks Rangkaian Kontrol.

Pembuatan boks untuk motor DC sebagai penggerak kincir air diharapkan untuk pijakan motor dc dan juga sebagai pelindung yang diharapkan bisa menahan disaat cuaca sedang hujan supaya motor tidak terkena air hujan mengingat letak kolam yang dilapangan terbuka. Pembuatan boks untuk tempat motor dc sama halnya dengan boks untuk rangkaian control yaitu memakai bahan akrilik yang transparan dengan ketebalan 3mm panjang 20 cm, lebar 6.5 cm dan tinggi 13.5 cm seperti Gambar 26.



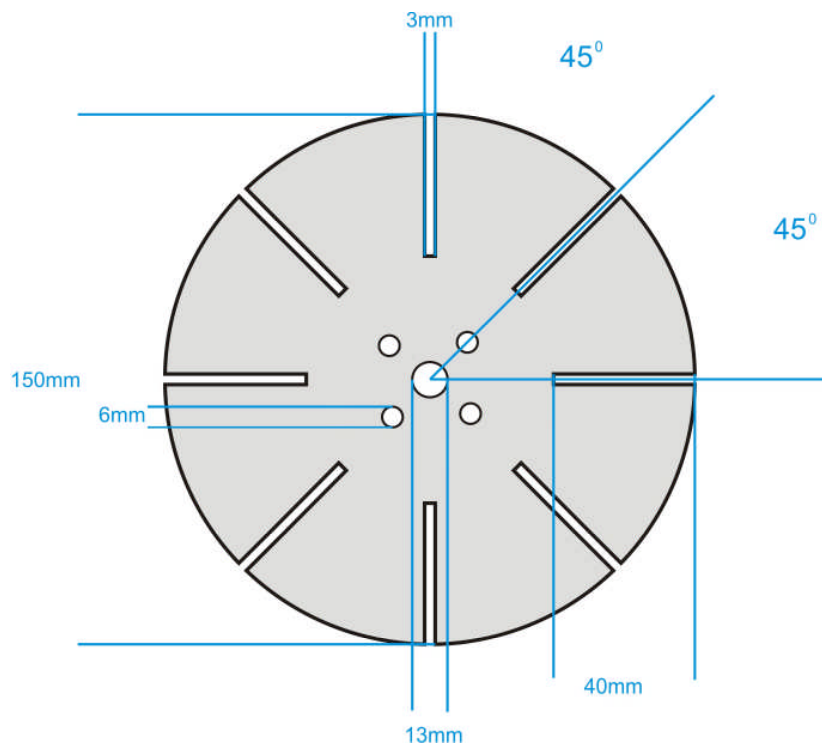
Gambar 26. Desain Boks Motor DC

#### D. Perancangan Sistem Mekanis

Konstruksi mekanis dari Kincir Air Alternatif dengan Timer sebagai Penyuplai Kandungan Oksigen (*dissolved oxygen*) pada Kolam Pembenihan Ikan Lele Berbasis Mikrokontoler ATmega8 ini menggunakan bahan dari akrilik. Dipilih dari bahan akrilik karena melihat dari bahannya yang ringan dan sudah ada tempat yang khusus membentuk dan memotong sesuai keinginan. Selain itu tidak terpengaruh terhadap ikan dikarenakan sudah adanya akuarium yang memakai bahan akrilik.

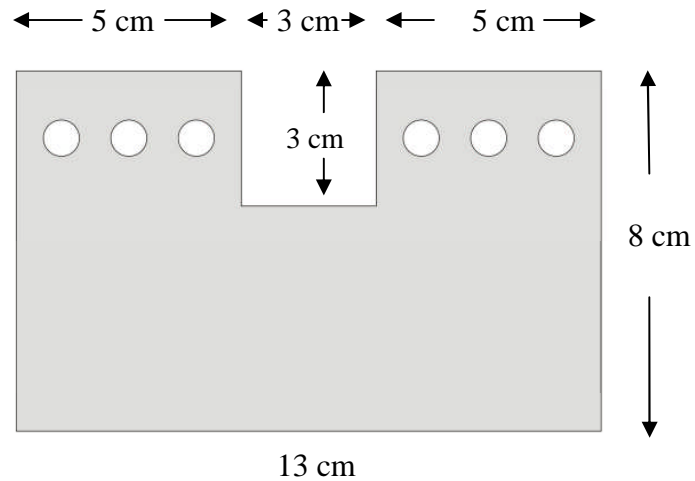
##### a. Perancangan konstruksi kincir air.

Pembuatan konstruksi kincir air menggunakan bahan akrilik dengan tebal 1 cm dan 3 mm. Pemotongan menggunakan potong laser. Gambar 27 berikut ini adalah gambar desain kincir air.



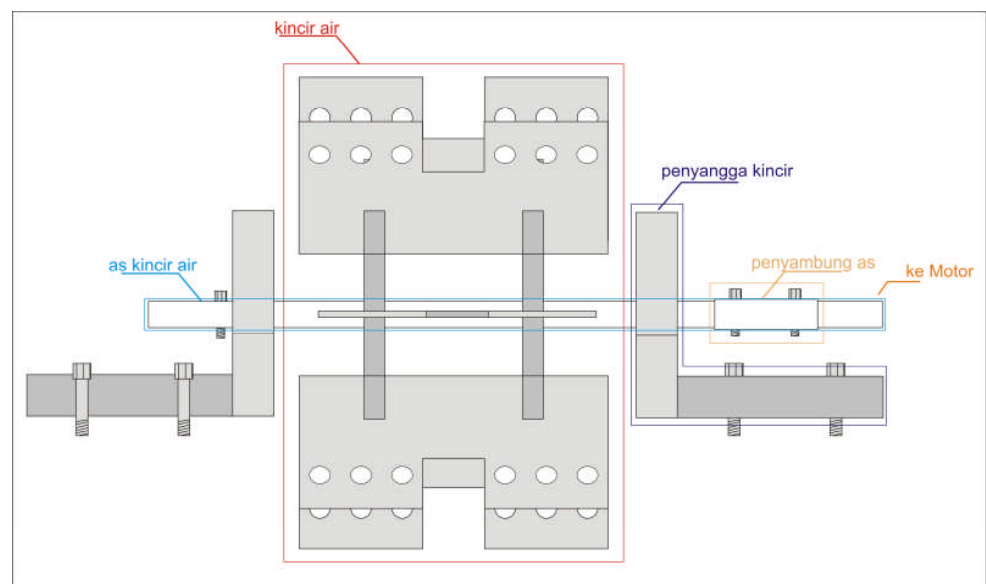
Gambar 27. Desain Kincir Air

Dalam kincir air tentu terdapat sirip – sirip yang fungsinya untuk mengoyak air sehingga timbul gelombang – gelombang pada air. Desain sirip – sirip kincir air dapat dilihat pada Gambar 28 berikut:



Gambar 28. Sirip – sirip Kincir Air

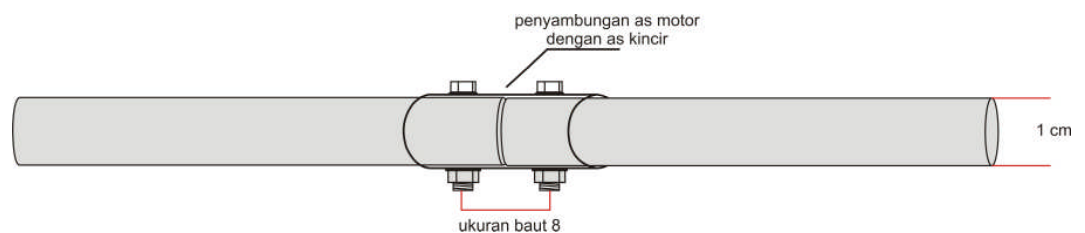
Gambar 29 ini menunjukkan bagian – bagian desain jadi perancangan kincir air yang akan digunakan.



Gambar 29. Desain Jadi Kincir Air

b. Perancangan as penghubung motor dengan kincir

Pembuatan as sebagai penghubung antara motor DC sebagai penggerak kincir air menggunakan bahan alumunium pejal berbentuk lingkaran dengan diameter 1 cm. dipilih bahan alumunium dikarenakan sifatnya alumunium yang lebih ringan dibanding besi, lebih tahan karat karena tidak menutup kemungkinan as akan terkena air dari kolam, dan juga lebih murah daripada kuningan yang juga lebih berat *massanya*. Penyambungan dari as penghubung motor dengan kincir air dapat dilihat pada Gambar 30.

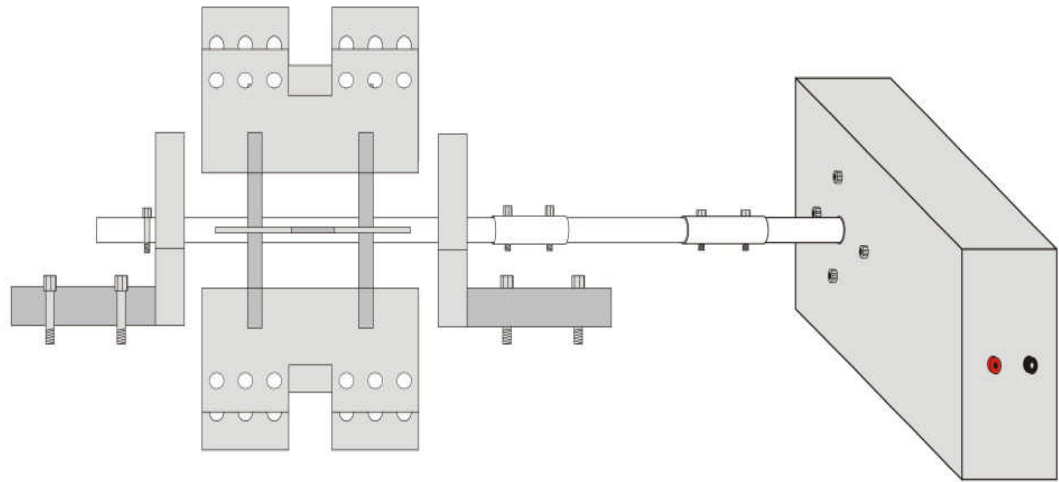


Gambar 30. Penyambungan As Penghubung Motor Ke Kincir Air

c. Desain Mekanik Secara Keseluruhan

Desain mekanik kincir air meliputi dari boks motor yang dihubungkan kincir air dengan menggunakan as penghubung yang terbuat dari alumunium. Untuk lebih jelasnya perhatikan Gambar 31.





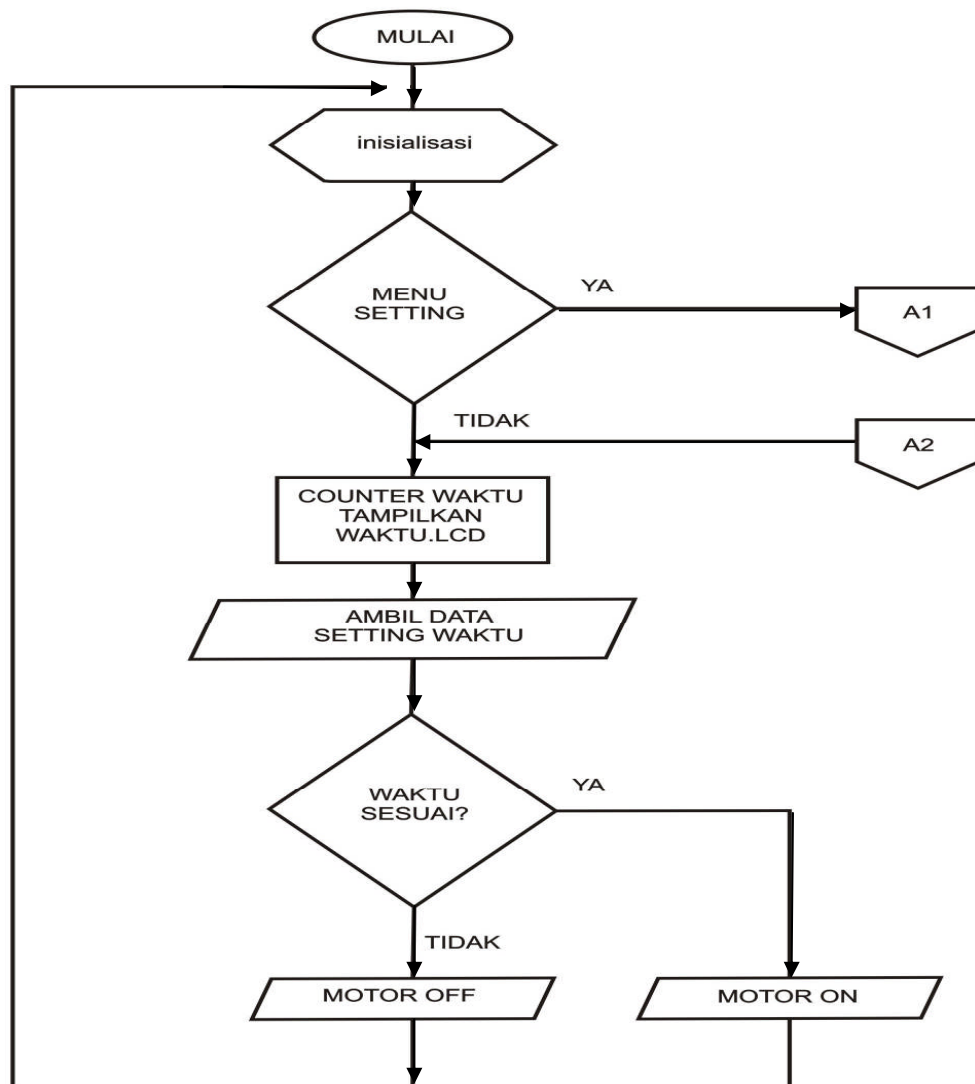
Gambar 31. Desain Mekanik

#### E. Perancangan Perangkat Lunak

Pada kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan ikan lele berbasis mikrokontroler ATmega8 diprogram menggunakan bahasa pemrograman C. Pemrograman dilakukan menggunakan *software Code Vision AVR*. Beberapa variabel yang merupakan data *setting* disimpan ke dalam EEPROM. Dengan demikian saat sistem tidak mendapatkan catu daya dan hendak digunakan kembali, pengguna (*user*) tidak perlu melakukan *setting* kembali. Program yang ditulis dengan bahasa C akan di-*compile*, kemudian file yang berekstensi \*.hex di-*download* ke dalam mikrokontroler.

Program kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan ikan lele berbasis mikrokontroler ATmega8 secara garis besar dapat digambarkan dalam diagram alir pada Gambar 32 sampai dengan Gambar 39.

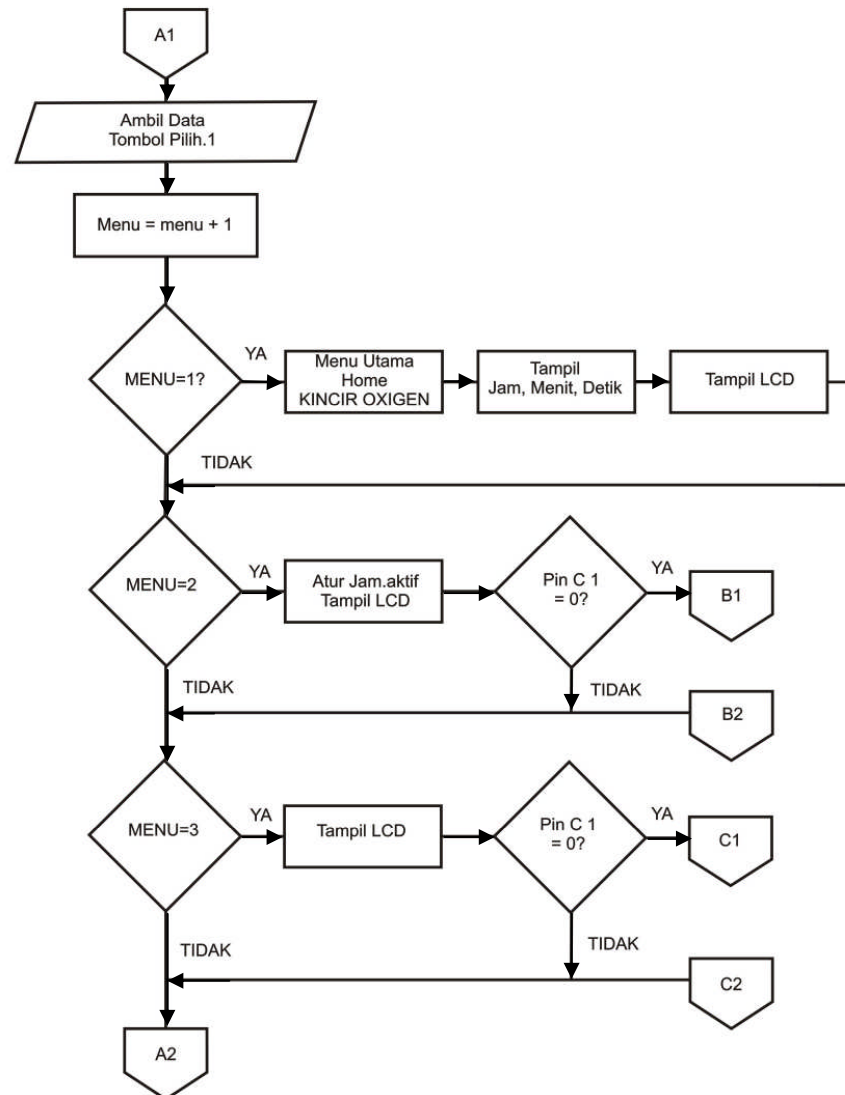
Inti dari diagram alir program utama dapat dilihat pada Gambar 32. Diagram alir program utama ini menjelaskan inti proses sistem kerja dari memulai sampai berakhirnya program.



Gambar 32. Diagram Alir Program Utama.

Fasilitas eksternal interup (*INT0*) digunakan untuk melakukan seting waktu (jam, menit, dan detik). Proses perpindahan dari satu menu ke menu

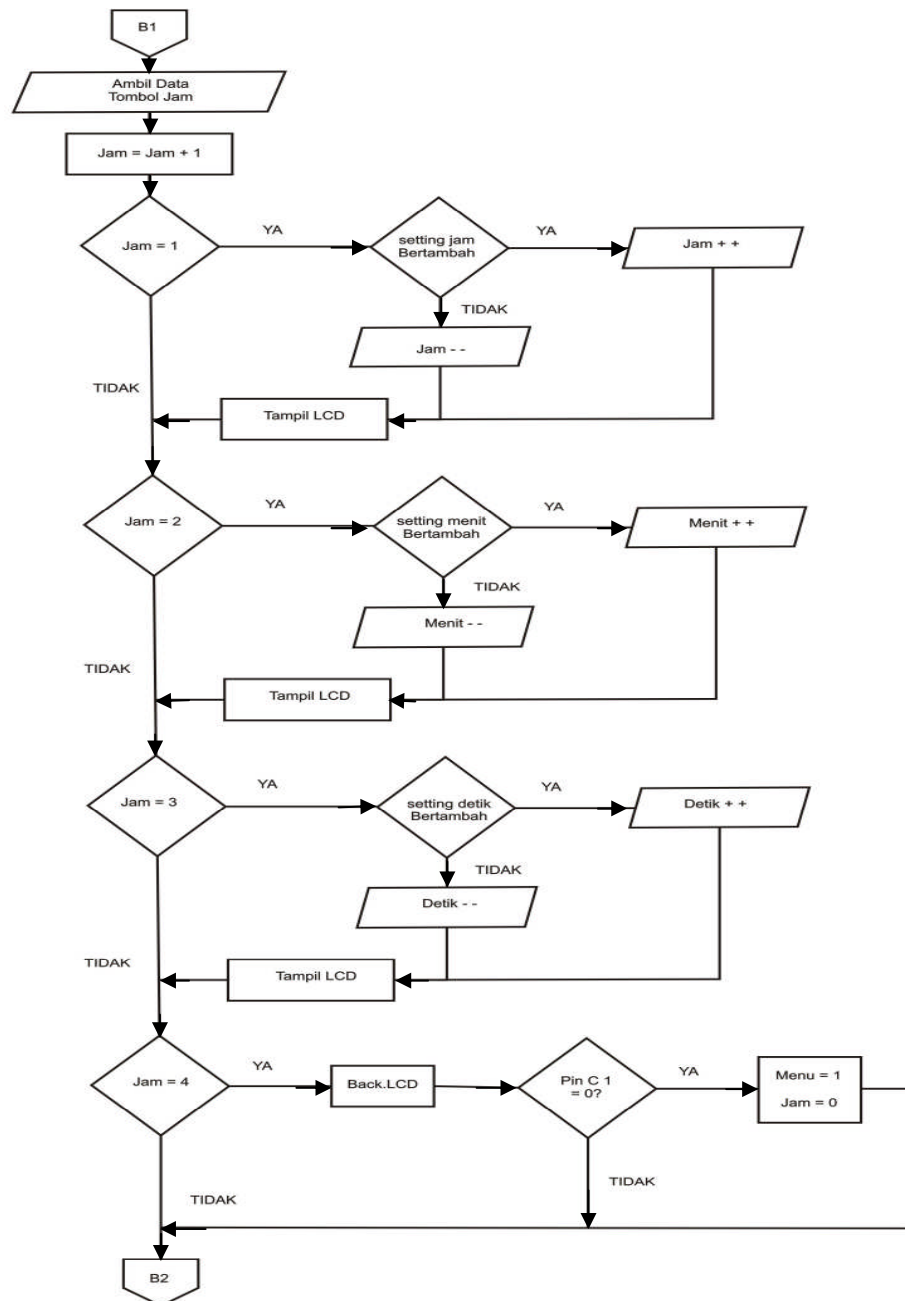
lainnya dilakukan dengan menekan tombol 1 secara berulang-ulang. Adapun penjelasan lengkapnya seperti pada Gambar 33 berikut:



Gambar 33. Diagram Alir Program *Setting* Menu

Proses dalam diagram *setting* menu ini adalah apabila tombol menu ditekan, maka tampilan akan berulang dari menu utama pada LCD ke menu

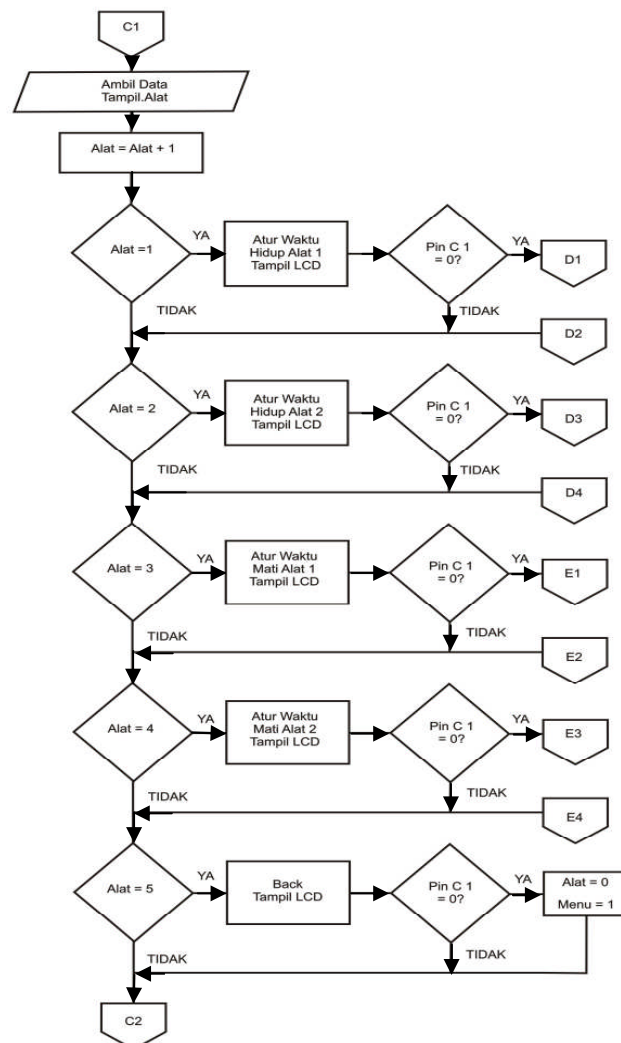
atur jam. Lebih jelasnya perhatikan Gambar 34. Program *setting* jam alat berfungsi untuk menyamakan antara jam sebenarnya dengan jam pada alat.



Gambar 34. Diagram Alir Program *Setting* Jam Alat.

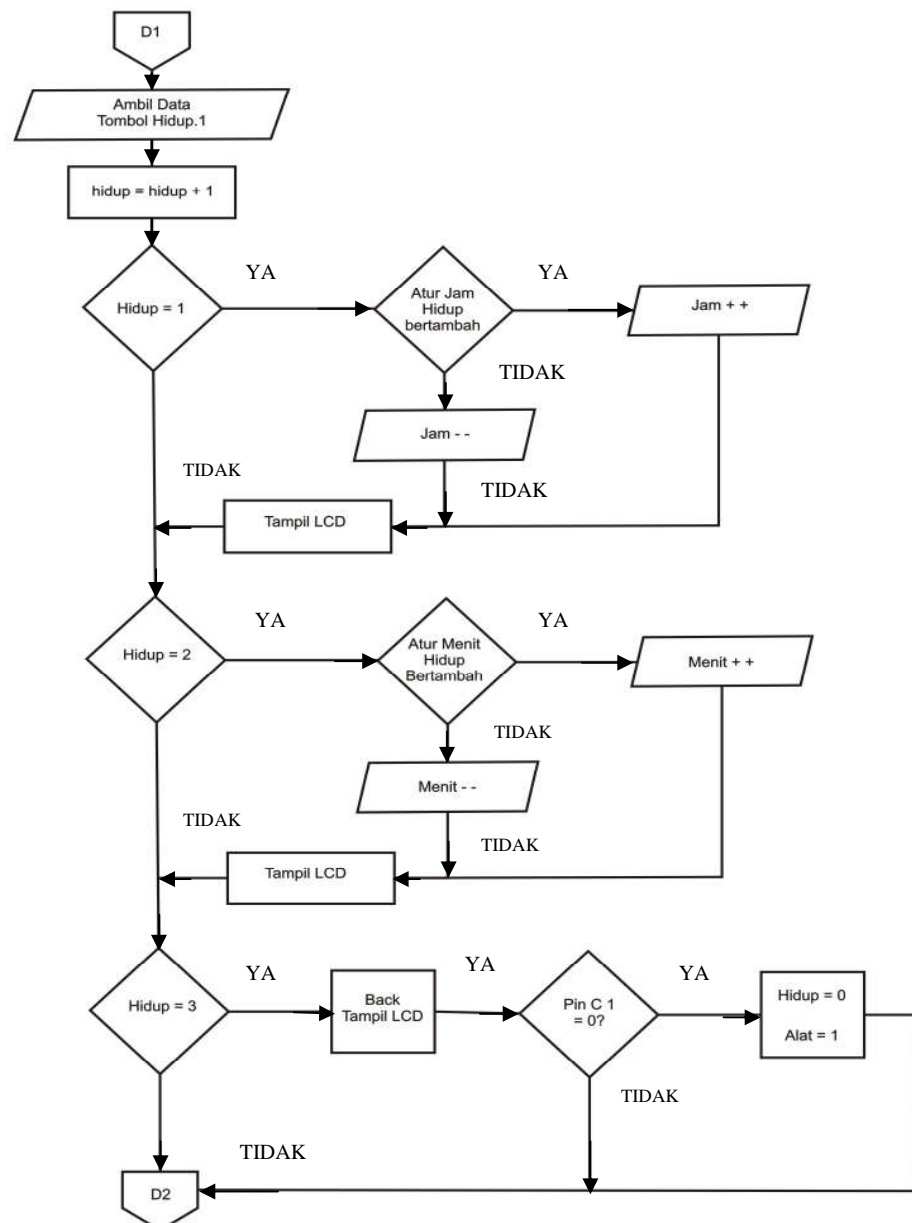
Dalam pengaturan jam aktif ini ada 3 pengaturan yang dilakukan yaitu pengaturan jam, menit dan detik. Disetiap pengaturan, baik itu pengaturan jam, menit maupun detik, apabila tombol + atau – ditekan, maka jam, menit maupun detik akan berjalan baik itu menambahkan ataupun mengurangi.

Perhatikan Gambar 35, Diagram alir program *setting* alat ini berisi pengaturan antara waktu menghidupkan alat, waktu mematikan alat dan juga menu *back* untuk kembali ke menu awal program.



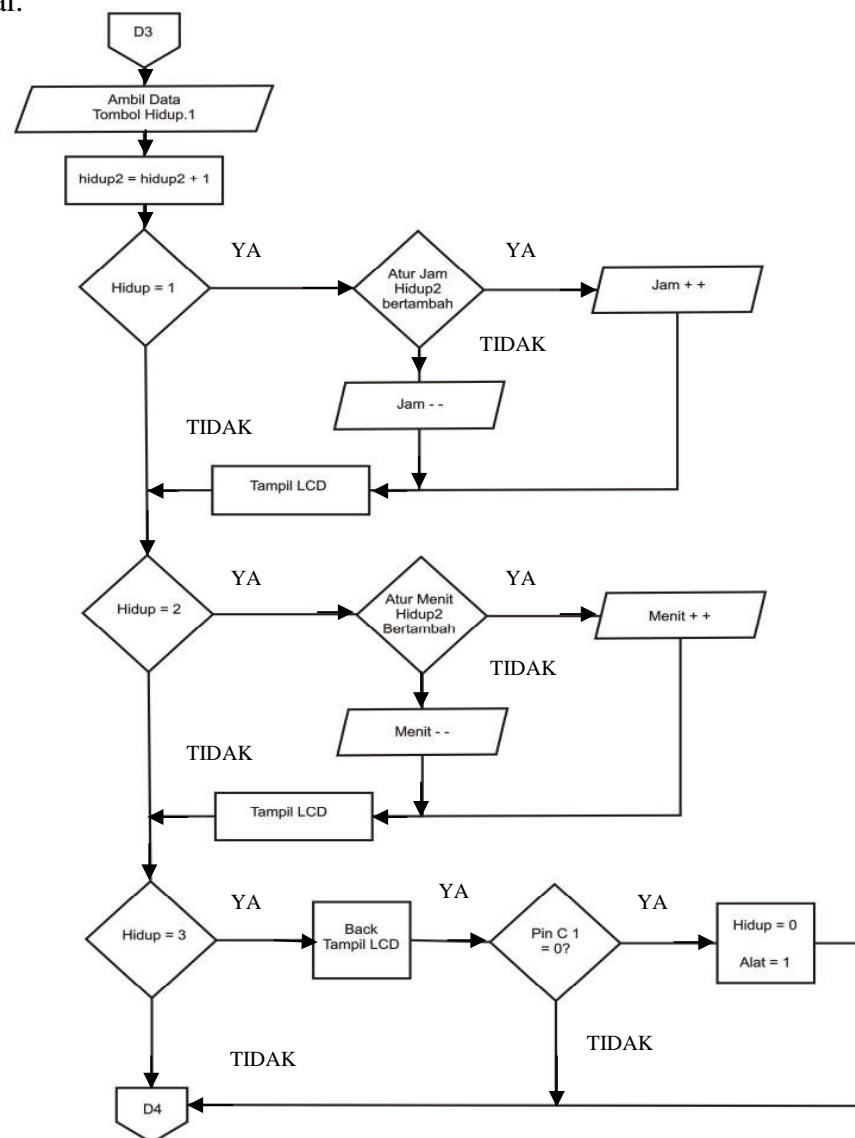
Gambar 35. Diagram Alir Program *Setting* Alat.

Diagram alir program *setting* hidup alat 1 ini merupakan sub dari program sebelumnya yang sudah dijelaskan pada Gambar 35. Pada pada *setting* hidup alat 1 ini menjalankan program untuk menghidupkan kincir air. Perhatikan Gambar 36.



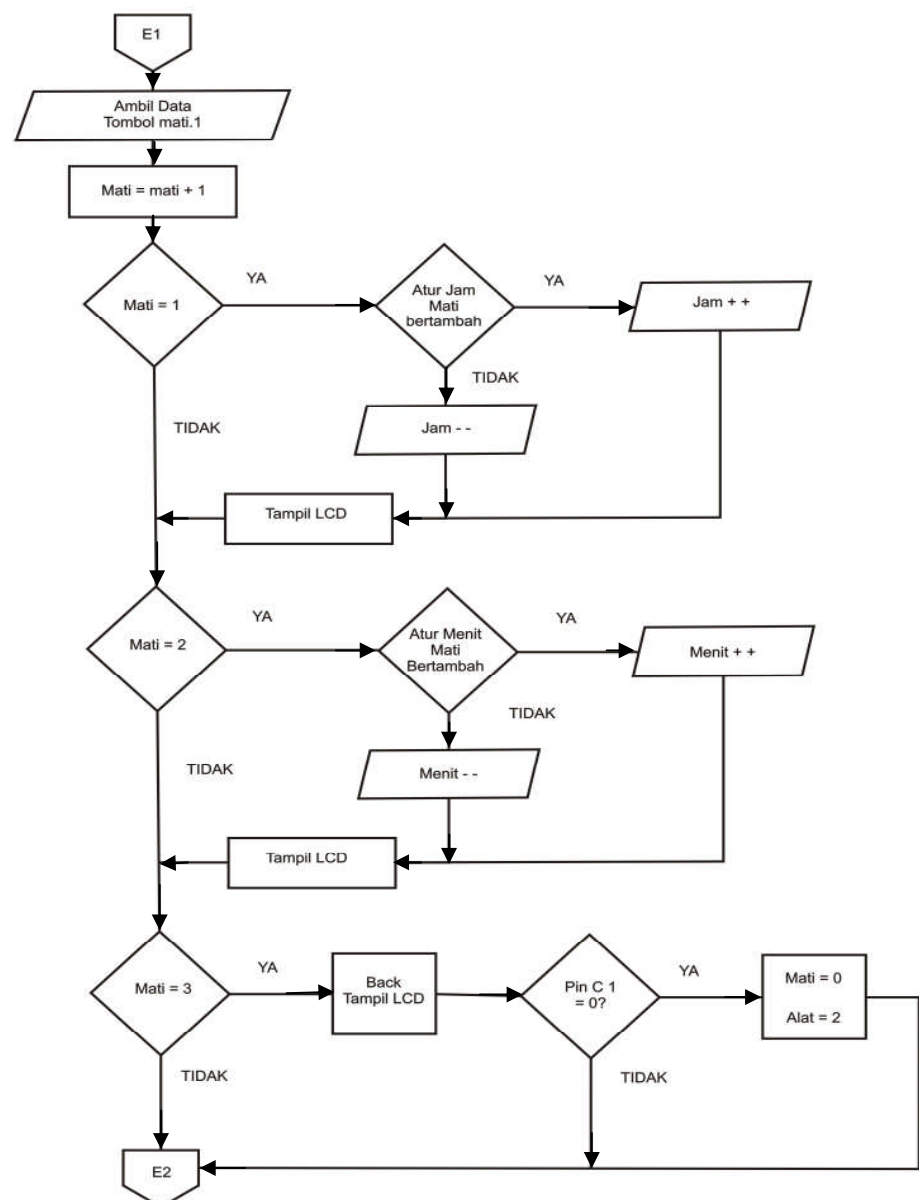
Gambar 36. Diagram Alir Program *Setting* Hidup Alat1.

Program *setting* hidup alat berfungsi untuk mengatur waktu kincir alat hidup pertama. Menu ini berisi atur jam hidup alat, menit hidup alat dan back yang ditampilkan pada LCD. Menu *back* berfungsi kembali ke menu sebelumnya. Selanjutnya mengatur *setting* hidup alat 2. Perhatikan Gambar 37, program *setting* hidup alat 2 ini berfungsi sama dengan *setting* hidup alat 1, tetapi program ini menghidupkan kincir air setelah proses *setting* alat 1 selesai.



Gambar 37. Diagram Alir Program *Setting* Hidup Alat2.

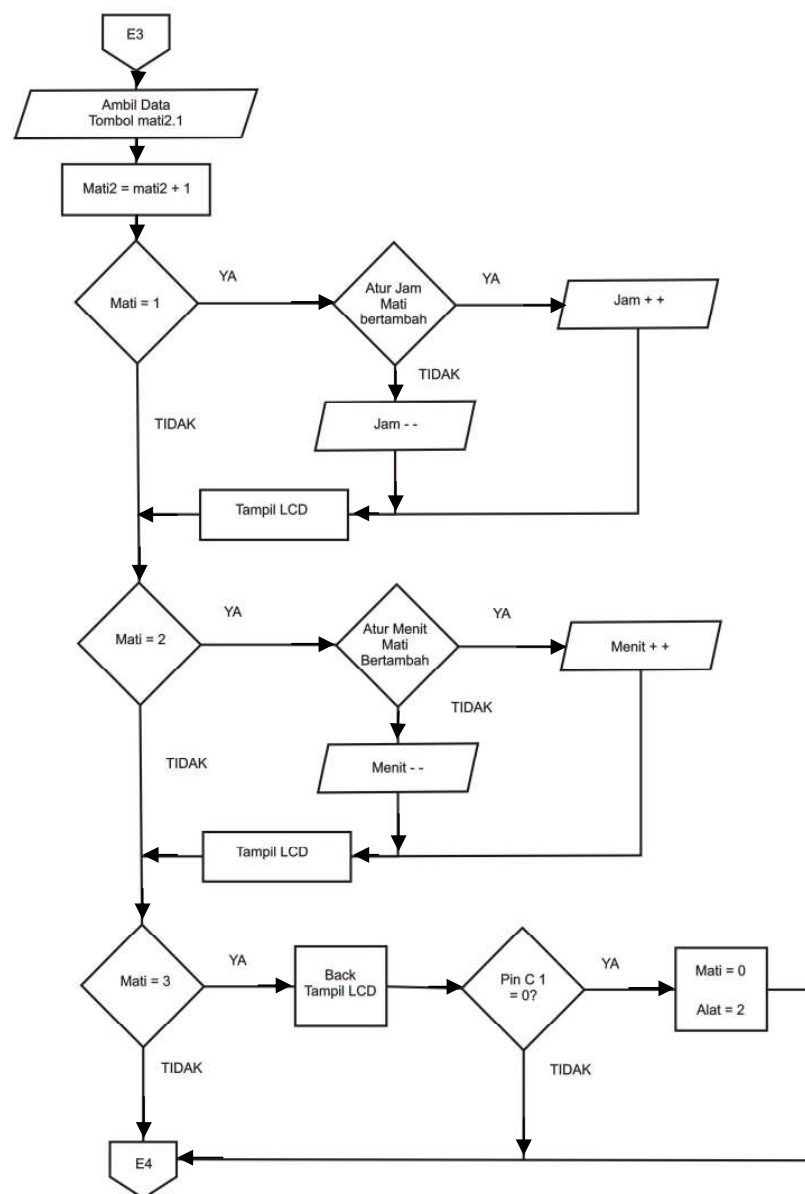
Gambar 38 menunjukkan program *setting* mati alat 1 ini cara kerjanya sama dengan *setting* hidup alat, namun program ini berfungsi untuk mematikan kincir. Pada alat nantinya program ini menjadi satu proses dengan *setting* hidup alat 1 pada Gambar 36.



Gambar 38. Diagram Alir Program *Setting* Mati Alat1.



Sama halnya seperti program setting mati alat1 pada Gambar 38, tetapi program ini berfungsi untuk mematikan program *setting* hidup alat 2. Dan program ini menjadi proses ke – 2 alat setelah proses ke – 1 dijalankan, yaitu dari proses menghidupkan alat 1 sampai proses mematikan alat 1.



Gambar 39. Diagram Alir Program *Setting* Mati Alat2.

## **F. Rencana Pengoperasian dan Pengujian Alat**

Pengoperasian kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan ikan lele berbasis mikrokontoler atmega8 dilakukan dengan menekan tombol setting. Selanjutnya kincir air akan hidup atau mati secara otomatis berdasarkan setting waktu yang telah ditentukan. Tombol setting terdiri dari 6 tombol setting, 1 diantaranya adalah tombol reset dan lima tombol berfungsi sebagai berikut:

- a. Tombol 1 (menu/ok) berfungsi untuk memilih menu seting ( setting jam, menit dan detik).
- b. Tombol 2 (geser kanan) berfungsi menggeser ke kanan untuk memilih pengaturan yang ingin diatur ( setting jam, setting kincir hidup atau setting kincir mati secara berulang).
- c. Tombol 3 (geser kiri) berfungsi sama seperti tombol 2 tetapi secara terbalik. Misal untuk tombol 2 urutannya setting jam, setting kincir hidup, setting kincir mati, pada tombol tiga dimulai dari setting kincir mati, setting kincir hidup dan setting jam.
- d. Tombol 4 (+/penambah) berfungsi sebagai penambah pada saat menentukan pengaturan jam, menit ataupun detik pada tiap pengaturan baik setting jam, setting kincir hidup dan setting kincir mati.
- e. Tombol 5 (-/pengurang) berfungsi sama seperti tombol 4 akan tetapi berfungsi untuk mengurangi pengaturan waktu baik pada saat menyetting jam, setting kincir hidup dan setting kincir mati.

- f. Tombol 6 berfungsi sebagai reset untuk mengulangi pengaturan waktu apabila terjadi gangguan pada saat menyetting alat.

1. Alat yang digunakan untuk pengujian dan pengambilan data adalah:

a. Kolam ikan

Kolam ikan digunakan untuk mengetahui fungsi alat dapat bekerja seperti yang diharapkan.

b. Multimeter

Multimeter digunakan untuk mengukur tegangan kerja pada catu daya.

2. Lokasi Pengujian dan Pengambilan Data

Untuk pengujian kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan ikan lele berbasis mikrokontoler atmega8 dilakukan di kolam ikan lele milik petani ikan yaitu Yahya Hamitisna, S.Pi. dan Buyung Pramono, S.Pi. di desa Kepatihan, Bantul Yogyakarta. Sedangkan untuk pengambilan data dilakukan di kediaman penulis karena untuk pengambilan data dari alat tidak perlu memakai peralatan khusus yang tersedia dilaboratorium.

### 3. Prosedur pengoperasian alat adalah sebagai berikut:

Pastikan alat sudah terhubung dengan tegangan sumber (PLN). Posisikan saklar power pada posisi ON untuk menghidupkan alat. Tekan tombol 1 (Menu) untuk memilih seting waktu yang akan diatur. Kemudian tekan tombol 2 (geser kanan) atau tombol 3 (geser kiri) untuk menggeser pilihan menu yaitu : *Setting Jam*, *Setting Kincir hidup*, *Setting Kincir Mati*. Apabila sudah ditentukan menu mana yang akan diatur, tekan kembali menu untuk pengaturan berikutnya yaitu jam, menit dan detik dengan menekan tombol 2 dan tombol 3 untuk menggeser serta tombol 4 dan tombol 5 untuk menambah juga mengurangi pengaturan jam, menit detik.

#### Kasus 1:

##### 1. Menentukan *Setting Jam*

Tekan tombol 1 (menu), maka muncul dilayar “setting jam”, tekan tombol 1 (menu) kembali, dilayar muncul “JAM”, tekan tombol 4 (+) untuk mengatur waktu “jam” secara menambah (*increment*) atau tombol 5 (-) untuk mengurangi (*decrement*). Setelah selesai tekan tombol 2 (geser kanan) atau tombol 3 (geser kiri) untuk menentukan “menit” atau “detik” dan tekan kembali tombol 4 atau tombol 5 untuk menambah atau mengurangi. Setelah selesai tekan kembali menu sampai layar kembali seperti semula yaitu tampilan “jam dan Kincir Oksigen”

## 2. Menentukan *Setting* Kincir Hidup.

Tekan tombol 1 (menu), kemudian tekan tombol 2 atau tombol 3 untuk menggeser sampai muncul tulisan “KINCIR HIDUP” pada layar LCD. Tekan tombol 1 (menu) untuk masuk pengaturan “JAM, MENIT dan DETIK” kemudian tekan tombol 4 atau tombol 5 untuk mengurangi atau menambah baik JAM, MENIT dan DETIK pada setting Kincir Hidup. Setelah selesai tekan tombol 1 kembali sampai tampilan layar LCD seperti semula.

## 3. Menentukan *Setting* Kincir Mati.

Tekan tombol 1 (menu), kemudian tekan tombol 2 atau tombol 3 untuk menggeser sampai muncul tulisan “KINCIR MATI” pada layar LCD. Tekan tombol 1 (menu) untuk masuk pengaturan “JAM, MENIT dan DETIK” kemudian tekan tombol 4 atau tombol 5 untuk mengurangi atau menambah. Setelah selesai tekan tombol 1 kembali sampai tampilan layar LCD seperti semula.

4. Pengujian alat meliputi:

a. Rencana pengujian Trafo 15 V DC (Setelah dioda *bridge*)

Tabel 8. Rencana pengujian Trafo 15 V DC (Setelah dioda *bridge*).

No.	Percobaan	Hasil pengukuran(V)
1	1	
2	2	

b. Rencana pengujian rangkaian catu daya

Tabel 9. Rencana pengujian *Power Supply* 6 V.

No.	<i>Input Voltage</i> (volt)	<i>IC LM 7806</i> <i>Output Voltage</i> (volt)	<i>Tr 2N 3055</i> <i>Output Voltage</i> (volt)
1	9		
2	15		

Tabel 10. Rencana pengujian *Power Supply* 12 V.

No.	<i>Input Voltage</i> (volt)	<i>IC LM7812</i> <i>Output Voltage</i> (volt)
1	15	
2	15	
3	15	

c. Rencana pengujian rangkaian Saklar *Relay*Tabel 11. Rencana pengujian rangkaian Saklar *Relay*.

No.	<i>Driver Relay</i>	Logika <i>Input</i>	Keterangan
1	1	1	
		0	

## d. Rencana pengujian Fungsi Tombol

Tabel 12. Rencana pengujian fungsi tombol.

No	Tombol	Ket. Tombol	Penekanan Tombol	keterangan
1	1	Menu (OK)	1	
2	2	kanan	1	
3			2	
4			3	
5	3	kiri	1	
6			2	
7			3	
8	4	+	1	
9	5	-	1	

## e. Rencana pengujian alat

Tabel 13. Rencana Pengujian Kesesuaian Waktu Pada Alat

Percobaan ke	Waktu sesungguhnya	Waktu pada alat	Selisih Waktu
1			
2			
3			
4			
5			
6			
7			

Tabel 14. Rencana Pengujian Kesesuaian Waktu Setting Kincir Dengan

Unjuk Kerja Kincir Saat Nyala Dan Mati

No	Percobaan	Waktu Setting Hidup	Waktu Setting Mati	Waktu Hidup	Waktu Mati	Selisih Waktu Hidup	Selisih Waktu Mati
1	1						
2	2						
3	3						
4	4						
5	5						



## BAB IV

### HASIL DAN PEMBAHASAN

#### A. Pengujian

##### 1. Tujuan

Pengujian ini bertujuan untuk mengetahui kebenaran rangkaian dan mengetahui kondisi komponen, perangkat lunak, serta unjuk kerja *kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (dissolved oxygen) pada kolam pembenihan lele berbasis mikrokontroler ATmega8* secara keseluruhan. Pengambilan data dilakukan dengan melakukan pengukuran pada masing-masing blok sistem ataupun komponen yang digunakan sehingga dapat dihasilkan perbandingan antara teori dan secara prakteknya.

##### 2. Tempat dan Waktu Pengambilan Data

Pengambilan data dilakukan di kediaman penulis dan pengujian alat pada kolam ikan pembenihan ikan lele di desa kepatihan bantul pada bulan Mei 2011.

#### B. Hasil Pengujian Alat

##### 1. Catu Daya (*Power Supply*)

Pada rangkaian *power supply* terdiri dari *transformator step down* sebagai penurun tegangan, dioda *bridge* sebagai penyearah gelombang penuh, kapasitor sebagai *filter*, IC regulator tegangan sebagai penyetabil

tegangan, dan transistor sebagai penguat daya. Rangkaian catu daya yang dibuat tersebut menghasilkan tegangan +5,3 V dan +12 V. Untuk mendapatkan tegangan yang stabil sesuai dengan yang diinginkan, maka digunakan IC regulator tegangan IC LM7806 yang dilengkapi dengan transistor daya 2N3055 yang digunakan untuk menghasilkan tegangan sebesar +5,3 V DC juga IC LM7812 yang digunakan untuk mendapatkan tegangan sebesar +12 V. Pengamatan dilakukan dengan mengukur tegangan *power supply* menggunakan multimeter. Pengamatan tersebut menghasilkan tegangan yang hampir sama dengan tegangan keluaran yang diinginkan.

Tabel 15 . Hasil Pengujian Trafo 15 V Dc (Setelah Dioda *Bridge*) dengan Menggunakan Multimeter

No.	Percobaan	Hasil pengukuran(V)
1	1	18.1
2	2	18.4

Hasil pengukuran pada catu daya menunjukkan bahwa :

Tegangan keluaran trafo rata – rata :

$$V_{dc} = \frac{18.1 + 18.4}{2}$$

$$= 18.25 \text{ Volt}$$

Secara Teoritis :

- Tegangan keluaran trafo 15 Volt

Dimana persamaan tegangan keluaran DC rerata yaitu :

$$V_{dc} = \frac{2 [(V_{m \text{ trafo}} - (2 \times V_{dioda}))]}{\pi}$$

$$= 0,637 \times [V_{m \text{ trafo}} - (2 \times V_{dioda})]$$

Sehingga dari persamaan diatas diperoleh nilai tegangan DC rerata

$$V_{dc} = \frac{2 (15\sqrt{2} - 1.4)}{\pi}$$

$$= 0,637 \times [15\sqrt{2} - 1.4]$$

$$= 12.62 \text{ Volt}$$

Tabel 16. Hasil Pengujian *Power Supply* 6 V

No.	Input Voltage (volt)	IC LM 7806 Output Voltage (volt)	Tr 2N 3055 Output Voltage (volt)
1	9	5.8	5.3
2	15	5.78	5.3

$$V_{\text{out rata-rata LM7806}} = \frac{(5.8 + 5.78)}{2} = 5.79 \text{ volt}$$

$$V_{\text{out rata-rata 2N 3055}} = \frac{(5.3 + 5.3)}{2} = 5.3 \text{ volt}$$

Perbedaan hasil tegangan antara keluaran trafo setelah dioda dengan tegangan *power supply* 6 volt maupun 12 volt terlihat sangat mencolok. Ini disebabkan tegangan *power supply* baik 6 vol maupun 12

volt sudah melewati *IC* regulator tegangan yang berfungsi untuk menyetabilkan tegangan keluaran pada *power supply*.

Berdasarkan hasil pengujian *power supply* 6 volt, besarnya tegangan keluaran IC LM7806 adalah 5.79 volt. Idealnya, besar tegangan keluaran IC LM7812 adalah 6 volt.

Penyimpangan tegangan keluaran sebesar:

$$\begin{aligned}\text{Penyimpangannya sebesar} &= \frac{6 - 5.79}{6} \times 100 \% \\ &= 3.5 \%\end{aligned}$$

Penyimpangan yang terjadi cukup kecil yaitu sebesar 3.5 %. Penyimpangan tersebut masih dalam toleransi. Tegangan keluaran ini digunakan untuk mencatu rangkaian sistim minimum.

Tabel 17. Hasil pengujian power suplay 12 volt

No.	<i>Input Voltage</i> (volt)	<i>IC LM7812</i> <i>Output Voltage</i> (volt)
1	15	12.1
2	15	12.4
3	15	12.2

$$V_{\text{out rata-rata LM7812}} = \frac{(12.1 + 12.4 + 12.2)}{3}$$

$$V_{\text{out rata-rata LM7812}} = 12.23 \text{ volt}$$

Berdasarkan hasil pengujian *power supply*, besarnya tegangan keluaran IC LM7812 adalah 12.23 volt. Idealnya, besar tegangan keluaran IC LM7812 adalah 12 volt.

Penyimpangan tegangan keluaran sebesar:

$$\begin{aligned}\text{Penyimpangannya sebesar} &= \frac{12.23 - 12}{12} \times 100 \% \\ &= 1.91 \%\end{aligned}$$

Penyimpangan yang terjadi cukup kecil yaitu sebesar 1.91 %. Penyimpangan tersebut masih dalam toleransi. Tegangan keluaran ini digunakan untuk mencatu rangkaian motor dc untuk menggerakkan kincir air.

## 2. Sistem *Driver Relay*

Rangkaian saklar *relay* dirancang menggunakan *relay* DC 5 volt. Apabila rangkaian mendapatkan logika 1 (*high*), maka kontak *NO* di dalam *relay* akan terhubung. Sedangkan apabila rangkaian mendapatkan logika 0 (*low*), kontak *NO* di dalam *relay* akan terputus. Untuk mengontrol *relay* digunakan rangkaian *driver*.

Rangkaian *driver relay* dirancang dengan menggunakan transistor. Rangkaian ini memanfaatkan transistor sebagai saklar elektronis yang dapat menghidupkan dan mematikan *relay*. Setelah mencapai tegangan kerja, transistor akan berfungsi sebagai saklar tertutup. Saat demikian, kumparan *relay* mendapatkan arus listrik. Inti

besi akan menjadi magnet dan menarik kontak *relay*, sehingga kontak *NO* menjadi terhubung, dan kontak *NC* menjadi terputus. Pengujian *driver relay* ini memanfaatkan *output* dari mikrokontroler, dimana *output* dari mikrokontroler ini akan menjadi *input* dari rangkaian *driver relay*. Hasil dari pengujian *driver relay* ditunjukkan pada Tabel 18 berikut:

Tabel 18. Hasil Pengujian Rangkaian *Driver Relay*

No.	<i>Driver Relay</i>	Logika <i>Input</i>	Keterangan
1	1	1	ON
		0	OFF

### 3. Tombol *Setting*

Dari pengujian tombol seting, didapatkan hasil yang memuaskan. Hal ini dikarenakan semua tombol berfungsi sesuai dengan yang diharapkan. Adapun uraian fungsi masing-masing tombol adalah seperti pada Tabel 19.

Tabel 19. Hasil Pengujian fungsi tombol.

No	Tombol	Ket. Tombol	Penekanan Tombol	keterangan
1	1	Menu (OK)	1	Untuk mengeksekusi program.
2	2	kanan	1	Menu setting jam
3			2	Menu setting alat
4			3	Kembali kehalaman awal
5	3	kiri	1	Menu setting alat
6			2	Menu setting jam
7			3	Kembali kehalaman awal
8	4	+	1	Jam ++, menit ++, detik ++
9	5	-	1	Jam --, menit --, detik --

#### 4. Pengujian alat

Pengujian alat ini berfungsi untuk mengetahui selisih waktu pada alat dengan waktu sebenarnya dengan jam tangan dan juga untuk mengetahui selisih waktu hidup alat dan waktu mati alat. Hasilnya tidak ada perbedaan selisih waktu yang signifikan antara waktu pada alat dengan waktu sebenarnya. Tabel 20 berikut adalah hasil yang didapat dari percobaan.

Tabel 20. Hasil Pengujian Kesesuaian Waktu Antara Waktu

Sesungguhnya dengan Waktu Pada Alat.

Percobaan ke	Waktu sesungguhnya	Waktu pada alat	Selisih Waktu
1	18:00:00	18:00:00	00:00:00
2	18:00:00	18:00:00	00:00:00
3	19:30:00	19:30:00	00:00:00
4	20:00:00	20:00:00	00:00:00
5	21:30:00	21:30:00	00:00:00
6	22:00:00	22:00:00	00:00:00
7	23:30:00	23:30:00	00:00:00

Tabel 21. Hasil Pengujian Kesesuaian Waktu Setting Kincir

Dengan Unjuk Kerja Kincir Saat Nyala Dan Mati

No	Percobaan	Waktu Setting Hidup	Waktu Setting Mati	Waktu Hidup	Waktu Mati	Selisih Waktu Hidup	Selisih Waktu Mati
1	1	08:00:00	08:05:00	08:00:00	08:05:00	00:00:00	00:00:00
2	2	09:00:00	09:10:00	09:00:00	09:10:00	00:00:00	00:00:00
3	3	10:30:00	10:45:00	10:30:00	10:45:00	00:00:00	00:00:00
4	4	13:00:00	13:20:00	13:00:00	13:20:00	00:00:00	00:00:00
5	5	14:30:00	15:00:00	14:30:00	15:01:00	00:00:00	00:00:01

### C. Pembahasan

#### 1. Catu daya

Catu daya terdiri atas transformator step down (penurun tegangan), diode bridge (*bridge rectifier*), kapasitor, IC regulator tegangan seri LM 7806 dan LM 7812 serta transistor 2N3055.

Transformator ini berfungsi untuk menurunkan tegangan dari 220 V AC menjadi 15 V AC. Sedangkan dioda bridge berfungsi untuk menyearahkan atau mengubah tegangan AC menjadi DC dengan gelombang penuh yang dipasang pada sisi sekunder trafo yaitu pada keluaran 15 V trafo.

##### a. Tegangan keluaran transformator 15 Volt

Dimana persamaan tegangan keluaran DC rerata yaitu :

$$V_{dc} = \frac{2 [(V_m \text{ trafo} - (2 \times V_{dioda}))]}{\pi}$$

$$= 0,637 \times [V_{m\text{trafo}} - (2 \times V_{dioda})]$$

Sehingga dari persamaan diatas diperoleh nilai tegangan DC rerata

$$V_{dc} = \frac{2 (15\sqrt{2} - 1.4)}{\pi}$$

$$= 0,637 \times [15\sqrt{2} - 1.4]$$

$$= 12.62 \text{ Volt}$$



Meskipun sudah mencukupi untuk masukan regulator tegangan seri LM 7812, tegangan DC hasil penyearahan dari *Bridge Rectifier* ini tidaklah rata. Untuk meratakannya perlu dipasang tapis kapasitor. Pemasangan tapis kapasitor juga akan meningkatkan nilai tegangan rerata. Pengosongan muatan yang lambat pada kapasitor akan mempertahankan nilai tegangan pada tegangan puncak  $15\sqrt{2}$  V sehingga riak tegangan berkurang. Berikut persamaan tegangan DC setelah dipasang kapasitor :

$$V_{dc} = [V_{m_{trafo}} - (2 \times V_{dioda})] - \frac{V_r}{2}$$

$$\begin{aligned} \text{Dimana, } V_r &= \frac{V_{dc}}{2.f.C} \\ &= \frac{V_{dc}}{2.f.C. R} \end{aligned}$$

$$\text{Sehingga, } V_{dc} = [15\sqrt{2} - 1.4] - \frac{V_{DC}}{4 \times 50 \times 2200^{-6} \times 28}$$

$$\begin{aligned} V_{dc} &= 19.81 - \frac{V_{dc}}{17.6} \\ V_{dc} + \frac{V_{dc}}{17.6} &= 19.81 \text{ volt} \end{aligned}$$

$$\frac{17.6V_{dc} + V_{dc}}{17.6} = 19.81 \text{ volt}$$

$$\frac{18.6V_{dc}}{17.6} = 19.81 \text{ volt}$$

$$V_{dc} = \frac{19.81 \times 17.6}{18.6} \text{ volt}$$

$$V_{dc} = 18.74 \text{ Volt}$$

Penyimpangan keluaran sebesar :

$$\text{Kesalahan (error)} = \frac{18.74 - 18.23}{18.74} \times 100\% = 2.72\%$$

Penyimpangan yang terjadi cukup kecil yaitu sebesar 2.72%. Hal itu masih dapat diabaikan dan lebih aman karena selisih 2.72% dari tegangan keluaran idealnya.

b. Tegangan keluaran IC LM 7806

Pada hasil pengujian tegangan keluaran IC LM 7806 didapati nilai dari tegangan keluaran rata – ratanya yaitu 5.79 volt DC. Idealnya, regulator akan mengeluarkan tegangan 6 volt DC.

Penyimpangan keluaran sebesar :

$$\begin{aligned} \text{Penyimpangannya sebesar} &= \frac{6 - 5.79}{6} \times 100\% \\ &= 3.5\% \end{aligned}$$

Penyimpangan yang terjadi cukup kecil yaitu sebesar 3.5 %. Hal tersebut masih dapat diabaikan dan dianggap aman karena selisih 3.5 % dari tegangan keluaran idealnya.

c. Tegangan keluaran IC LM 7812

Pada hasil pengujian tegangan keluaran IC LM 7812 didapati nilai dari tegangan keluaran rata – ratanya yaitu 12.23 volt DC. Idealnya, regulator akan mengeluarkan tegangan 12 volt DC.

Penyimpangan keluaran sebesar :

$$\begin{aligned}\text{Penyimpangannya sebesar} &= \frac{12.23 - 12}{12} \times 100 \% \\ &= 1.91 \%\end{aligned}$$

Penyimpangan yang terjadi cukup kecil yaitu sebesar 1.91%. Hal tersebut masih dapat diabaikan dan dianggap aman karena selisih 1.91 % dari tegangan keluaran idealnya.

## 2. Unjuk Kerja Sistem

Ketepatan *setting* waktu

Unjuk kerja alat ini dapat dilihat pada tabel 20 dan tabel 21. Tabel 20 yaitu pengujian ketepatan waktu antara waktu sebenarnya dengan waktu pada alat. Pengambilan data dilakukan dengan melakukan 7 kali percobaan pengambilan waktu. Pengambilan data tersebut didapati bahwa unjuk kerja ketepatan waktu seperti yang diharapkan.

Pada Tabel 21 berisi pengambilan data kesesuaian waktu setting alat dengan unjuk kerja alat saat kincir hidup dan mati. Ada satu perbedaan yang terjadi pada alat saat percobaan pengambilan data antara kesesuaian waktu pada alat dengan unjuk kerja saat alat hidup dan mati. Perbedaan tersebut tidaklah begitu signifikan dikarenakan perbedaannya hanya 1 detik saat mematikan alat.

## BAB V

### KESIMPULAN DAN SARAN

#### A. Kesimpulan

Setelah mengamati dan membahas “kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan lele berbasis mikrokontroler ATmega8”, maka didapatkan kesimpulan:

1. Kincir air alternatif dengan timer yang berfungsi sebagai penyuplai kandungan oksigen ini terdiri dari beberapa rangkaian penting antara lain rangkaian catu daya, driver relay dan sistem minimum. Sedangkan komponen yang dipakai yaitu motor DC, led dan tombol *setting*. Tombol *setting* berfungsi untuk mengatur setting waktu guna menjalankan motor DC yang nantinya berjalan otomatis.
2. Unjuk kerja dari kincir air dengan timer berbasis mikrokontroler ini telah menunjukkan hasil sesuai dengan yang diharapkan. Kincir air dapat bekerja apabila saklar relay mendapat logika 1 dari output mikrokontroler yang menjadi inputan untuk saklar relay. saklar relay akan menghidupkan motor dc dengan kecepatan rata – rata 100 rpm untuk memutar kincir air. Ketepatan proses alat dengan kebenaran 97.4%. Alat dapat bekerja mulai dari proses seting waktu, proses seting kincir on dan proses seting kincir off. Semua akan berjalan otomatis setelah proses seting waktu ditentukan.

## B. Keterbatasan Alat

Keterbatasan “kincir air alternatif dengan timer sebagai penyuplai kandungan oksigen (*dissolved oxygen*) pada kolam pembenihan lele berbasis mikrokontroler ATmega8” ini antara lain:

1. Alat ini tidak ditambahkan dengan *back up* catu daya sehingga pada saat sumber dari PLN mati maka pada saat sumber menyala harus mensetting waktu kembali untuk menjalankan kembali kincir air.
2. Perlu adanya pendesainan ulang antara penghubung motor dengan kincir air karena dengan sistem ini masih mempunyai kelemahan saat kincir hidup penghubung yang tidak presisi mengakibatkan bergeraknya motor. Terpikirkan oleh penulis untuk mengganti dengan sistem penghubung seperti gardan.
3. Tidak adanya alat pengukur kadar oksigen dalam air sehingga kadar oksigen dalam air tidak bisa ditampilkan.
4. Tidak adanya sensor pengukur kadar oksigen sehingga apabila kadar oksigen dirasa sudah cukup kincir mati dan sebaliknya.

### C. Saran

Berdasarkan kendala-kendala yang penulis alami saat penelitian, maka pada kesempatan ini penulis memberikan beberapa saran sebagai berikut:

1. Diharapkan pada pengembangan berikutnya dapat menambahkan *back up* catu data supaya pada saat sumber PLN mati setting waktu pada alat ini tetap berjalan dan tidak perlu mensetting ulang waktu sampai sumber PLN hidup.
2. Bisa ditambahkan dengan sensor kadar oksigen air supaya alat ini berjalan secara otomatis apabila kandungan oksigen dalam air sangatlah minim pada kolam, jadi terpacu hanya pada sensor kandungan oksigen untuk menghidupkan atau mematikan alat ini.
3. Sistem penghubung antara motor dengan kincir lebih baik memakai sistem gardan yang bisa diatur sudut kemiringan dari motor ke kincir.
4. Membuat sistem kendali kadar oksigen sehingga alat ini semakin meningkat daya manfaatnya.

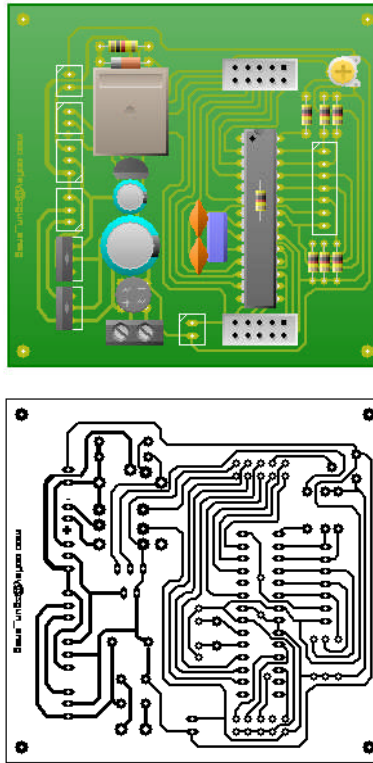
## DAFTAR PUSTAKA

- Andrianto, Heri. (2008). *Pemrograman Mikrokontroler AVR ATMEGA 8 Menggunakan Bahasa C (CodeVision AVR)*. Bandung: Informatika.
- Wardhana, Lingga. (2006). *Belajar Sendiri Mikrokontroler AVR Seri ATmega8 Simulasi, Hardware, dan Aplikasi*. Yogyakarta: Andi.
- Atmel Corporation.(2011). *Atmel 8-bit avr with 8K Bytes In-System Programmable Flash ATmega 8 ATmega 8L*".
- Sunomo. (1996) . *Elektronika II* . Yogyakarta: Teknik Elektro, Fakultas Teknik Universitas Negeri Yogyakarta.
- Ary Heryanto. (2007). *Pemrograman Bahasa C untuk Mikrokontroler ATMEGA8*. Yogyakarta: Andi Offset.
- Agus dkk. (2008). *Standar Petunjuk Operasional Tambak Boyo B*. Rogojampi: Banyuwangi.
- Ristek. (2009). *Sains & Teknologi*. Jakarta: Gramedia.
- Alphan. (2012). *Pengembangan Lampu Penerangan Jalan Menggunakan LED Dengan Pemanfaatan Sensor Cahaya Sebagai Pengontrol dan Timer Sebagai Back up Berbasis Mikrokontroller ATmega 8*. Proyek Akhir.
- Alam. (2012). *Processing Station Berbasis Mikrokontroler ATmega8*. Proyek Akhir.
- . *Data Sheet Komponen Elektronik*. Diakses dari [www.electroniclab.com](http://www.electroniclab.com) pada tanggal 15 November 2011, Jam 11.15 WIB.
- . *Datasheet Microcontroller ATmega8*. Diakses dari [www.atmel.com](http://www.atmel.com) pada tanggal 3 Maret 2012, Jam 15.16 WIB.

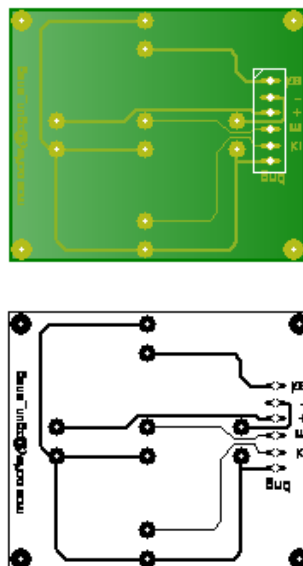


# **LAMPIRAN**

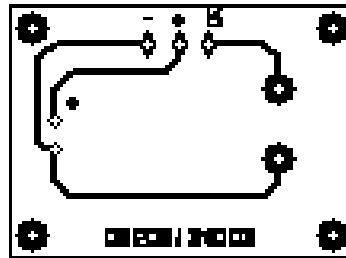
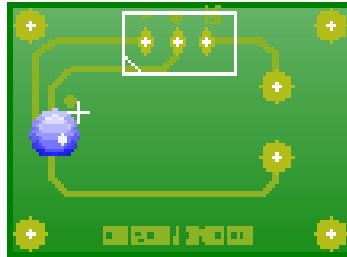
**Lampiran 1.** Gambar PCB Rangkaian Sistem Minimum dan Catu Daya



**Lampiran 2.** Gambar PCB Rangkaian Tombol *Setting*

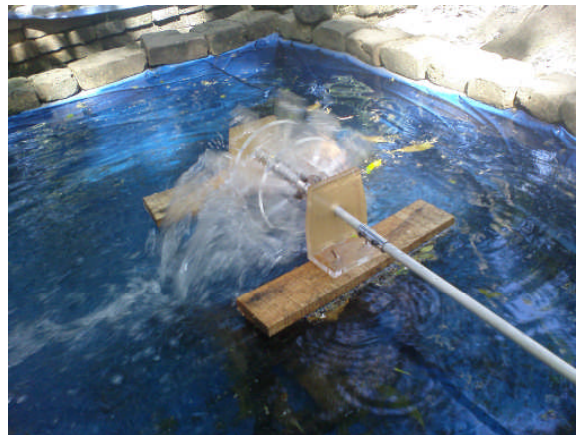


**Lampiran 3.** Gambar PCB Rangkaian Tombol *Reset* dan *LED*



**Lampiran 4.** Foto-Foto Proyek Akhir





### Lampiran 5. Program Mikrokontroller AVR ATmega8 Dengan Bahasa C

/\*\*\*\*\*\*

This program was produced by the  
CodeWizardAVR V2.05.3 Standard  
Automatic Program Generator  
© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.  
<http://www.hpinfotech.com>

Project : KINCIR AIR SEBAGAI PENYUPLAI OKSIGEN

Version : Rev.2

Date : 5/12/2012

Author : SRI ANGGANA A

Company : UNY

Comments: TUGAS AKHIR

Chip type : ATmega8

Program type : Application

AVR Core Clock frequency: 11.059200 MHz

Memory model : Small

External RAM size : 0

Data Stack size : 256

\*\*\*\*\*/

```
#include <mega8.h>
```

```
#include <stdlib.h>
```

```
#include <delay.h>
```

```
#include <stdio.h>
```

```
// Alphanumeric LCD functions
```

```
#include <alcd.h>
```

```

int detik=0,
    menit=0,
    jam=0,
    menith=0,
    jamh=0,
    menitm=0,
    jamm=0,
    menith1=0,
    jamh1=0,
    menitm1=0,
    jamm1=0;

unsigned char cdetik[10],
    cmenit[10],
    cjam[10],
    cmenith[10],
    cjamh[10],
    cmenitm[10],
    cjamm[10],
    cmenith1[10],
    cjamh1[10],
    cmenitm1[10],
    cjamm1[10];

int pilihset=1,
    pilihsetalat=0,
    a=0,
    b=0,
    alat=0,
    jamb=0,
    waktu_on=0,
    set_on=0,
    waktu_off=0,
    set_off=0;

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
    // Reinitialize Timer 1 value
    TCNT1H=0xD5;
    TCNT1L=0xD0;
    // Place your code here

    if
    ((PINC.1==1)&&(PINC.2==1)&&(PINC.3==1)&&(pilihset==1)&&(pilihsetalat==0))
    {

```

```

    if (++detik>=60)
    {
        detik=0;
        lcd_gotoxy(0,0);
        lcd_putsf("      ");
        if(++menit>=60)
        {
            menit=0;
            lcd_gotoxy(0,0);
            lcd_putsf("      ");
            if (++jam>=24)
            {
                jam=0;
                detik=2;
                lcd_gotoxy(0,0);
                lcd_putsf("      ");
            }
        }
    }

    if (((jam==jamh)&&(menit==menith))||((jam==jamh1)&&(menit==menith1)))
    {
        PORTC.5=1;
    }

    if
    (((jam==jamm)&&(menit==menitm))||((jam==jamm1)&&(menit==menitm1)))
    {
        PORTC.5=0;
    }
}
//menentukan MOTOR on/off
}

void main(void)
{
    // Crystal Oscillator division factor: 1
    #pragma optsize-
    CLKPR=0x80;
    CLKPR=0x00;
    #ifdef _OPTIMIZE_SIZE_
    #pragma optsize+
    #endif

    PORTB=0x00;
    DDRB=0x00;

```

```
PORTC=0x00;
DDRC=0x20;
```

```
PORTD=0x00;
DDRD=0xFF;
```

```
TCCR0A=0x00;
TCCR0B=0x00;
TCNT0=0x00;
OCR0A=0x00;
OCR0B=0x00;
```

```
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 10.800 kHz
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: On
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x05;
TCNT1H=0xD5;
TCNT1L=0xD0;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
```

```
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2A output: Disconnected
// OC2B output: Disconnected
ASSR=0x00;
TCCR2A=0x00;
TCCR2B=0x00;
TCNT2=0x00;
```

```
OCR2A=0x00;
OCR2B=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// Interrupt on any change on pins PCINT0-7: Off
// Interrupt on any change on pins PCINT8-14: Off
// Interrupt on any change on pins PCINT16-23: Off
EICRA=0x00;
EIMSK=0x00;
PCICR=0x00;

// Timer/Counter 0 Interrupt(s) initialization
TIMSK0=0x00;

// Timer/Counter 1 Interrupt(s) initialization
TIMSK1=0x01;

// Timer/Counter 2 Interrupt(s) initialization
TIMSK2=0x00;

// USART initialization
// USART disabled
UCSR0B=0x00;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
ADCSRB=0x00;
DIDR1=0x00;

// ADC initialization
// ADC disabled
ADCSRA=0x00;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
```



```

// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTD Bit 0
// RD - PORTD Bit 1
// EN - PORTD Bit 2
// D4 - PORTD Bit 4
// D5 - PORTD Bit 5
// D6 - PORTD Bit 6
// D7 - PORTD Bit 7
// Characters/line: 16
lcd_init(16);

// Global enable interrupts
#asm("sei")

while (1)
{
    if ((PINC.1==0)&&(a==0)&&(b==0)&&(jamb==0))
    {
        pilihset=pilihset-1;
        delay_ms(300);
    }

    if ((PINC.2==0)&&(a==0)&&(b==0)&&(jamb==0))
    {
        pilihset=pilihset+1;
        delay_ms(300);
    }

    if ((pilihset==2)&&(a==0)&&(b==0))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("SETTING JAM");
        delay_ms(100);
    }
    if ((pilihset==3)&&(a==0)&&(b==0))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("SETTING ALAT");
        delay_ms(100);
    }

    if ((pilihset==4)&&(a==0)&&(b==0))
    {

```

```

        pilihset=1;
        pilihsetalat=0;
    }
    if ((pilihset==0)&&(a==0)&&(b==0))
    {
        pilihset=3;
    }

    if((PINC.0==0)&&(pilihset==3)&&(alat==0))
    {
        a=1;
        alat=2;
        delay_ms(200);
    }

    if ((PINC.1==0)&&(a==1)&&(b==0)&&(waktu_on==0)&&(waktu_off==0))
    {
        alat=alat-1;
        delay_ms(300);
    }

    if ((PINC.2==0)&&(a==1)&&(b==0)&&(waktu_on==0)&&(waktu_off==0))
    {
        alat=alat+1;;
        delay_ms(300);
    }

    if ((a==1)&&(alat==2)&&(waktu_on==0)&&(waktu_off==0))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("WAKTU ON ALAT");
        delay_ms(100);
    }
    if ((a==1)&&(alat==3)&&(waktu_on==0)&&(waktu_off==0))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("WAKTU OFF ALAT");
        delay_ms(100);
    }
    if ((a==1)&&(alat==4)&&(waktu_on==0)&&(waktu_off==0))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("BACK");
    }

```

```

        delay_ms(100);
    }
    if ((a==1)&&(alat==5)&&(waktu_on==0)&&(waktu_off==0))
    {
        alat=2;
        delay_ms(100);
    }
    if ((a==1)&&(alat==1)&&(waktu_on==0)&&(waktu_off==0))
    {
        alat=4;
        delay_ms(100);
    }

    if
((PINC.0==0)&&(pilihset==3)&&(alat==4)&&(waktu_on==0)&&(waktu_off==0))
    {
        a=0;
        alat=0;
        pilihset=1;
        pilihsetalat=0;
    }

/*atur waktu oN alat*/

    if((PINC.0==0)&&(pilihset==3)&&(a==1)&&(alat==2)&&(set_on==0))
    {
        waktu_on=1;
        set_on=2;
        delay_ms(200);
    }

    if ((PINC.1==0)&&(waktu_on==1)&&(a==1)&&(alat==2))
    {
        set_on=set_on-1;
        delay_ms(300);
    }

    if ((PINC.2==0)&&(waktu_on==1)&&(a==1)&&(alat==2))
    {
        set_on=set_on+1;;
        delay_ms(300);
    }

    if ((waktu_on==1)&&(set_on==2))
    {
        lcd_clear();
    }

```

```

    lcd_gotoxy(0,1);
    lcd_putsf("JAM ON ALAT 1");
    lcd_gotoxy(0,0);
    lcd_puts(cjamh);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenith);
    delay_ms(100);
}
if ((waktu_on==1)&&(set_on==3))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("MENIT ON ALAT 1");
    lcd_gotoxy(0,0);
    lcd_puts(cjamh);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenith);
    delay_ms(100);
}
if ((waktu_on==1)&&(set_on==4))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("JAM ON ALAT 2");
    lcd_gotoxy(0,0);
    lcd_puts(cjamh1);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenith1);
    delay_ms(100);
}
if ((waktu_on==1)&&(set_on==5))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("MENIT ON ALAT 2");
    lcd_gotoxy(0,0);
    lcd_puts(cjamh1);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);

```

```

    lcd_puts(cmenith1);
    delay_ms(100);
}

if ((waktu_on==1)&&(set_on==6))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("BACK");
    lcd_gotoxy(0,0);
    lcd_puts(cjamh);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenith);
    lcd_gotoxy(11,0);
    lcd_puts(cjamh1);
    lcd_gotoxy(13,0);
    lcd_putsf(":");
    lcd_gotoxy(14,0);
    lcd_puts(cmenith1);

    delay_ms(100);
}

if ((waktu_on==1)&&(set_on==7))
{
    set_on=2;
}

if ((waktu_on==1)&&(set_on==1))
{
    set_on=6;
}

if ((PINC.0==0)&&(pilihset==3)&&(waktu_on==1)&&(set_on==6))
{
    a=1;
    alat=2;
    waktu_on=0;
    set_on=0;
    delay_ms(100);
}

/*atur waktu off alat*/

```

```

if((PINC.0==0)&&(pilihset==3)&&(a==1)&&(alat==3)&&(set_off==0))
{
    waktu_off=1;
    set_off=2;
    delay_ms(200);
}

if ((PINC.1==0)&&(waktu_off==1)&&(a==1)&&(alat==3))
{
    set_off=set_off-1;
    delay_ms(300);
}

if ((PINC.2==0)&&(waktu_off==1)&&(a==1)&&(alat==3))
{
    set_off=set_off+1;;
    delay_ms(300);
}

    if ((waktu_off==1)&&(set_off==2))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("JAM OFF ALAT 1");
        lcd_gotoxy(0,0);
        lcd_puts(cjamm);
        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenitm);
        delay_ms(100);
    }
    if ((waktu_off==1)&&(set_off==3))
    {
        lcd_clear();
        lcd_gotoxy(0,1);
        lcd_putsf("MENIT OFF ALAT 1");
        lcd_gotoxy(0,0);
        lcd_puts(cjamm);
        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenitm);
        delay_ms(100);
    }
    if ((waktu_off==1)&&(set_off==4))

```

```

{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("JAM OFF ALAT 2");
    lcd_gotoxy(0,0);
    lcd_puts(cjamm1);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenitm1);
    delay_ms(100);
}
if ((waktu_off==1)&&(set_off==5))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("MENIT OFF ALAT 2");
    lcd_gotoxy(0,0);
    lcd_puts(cjamm1);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenitm1);
    delay_ms(100);
}

if ((waktu_off==1)&&(set_off==6))
{
    lcd_clear();
    lcd_gotoxy(0,1);
    lcd_putsf("BACK");
    lcd_gotoxy(0,0);
    lcd_puts(cjamm);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenitm);

    lcd_gotoxy(11,0);
    lcd_puts(cjamm1);
    lcd_gotoxy(13,0);
    lcd_putsf(":");
    lcd_gotoxy(14,0);
    lcd_puts(cmenitm1);
    delay_ms(100);
}

```

```

if ((waktu_off==1)&&(set_off==7))
{
    set_off=2;
}
if ((waktu_off==1)&&(set_off==1))
{
    set_off=6;
}

if ((PINC.0==0)&&(pilihset==3)&&(waktu_off==1)&&(set_off==6))
{
    a=1;
    alat=3;
    waktu_on=0;
    set_on=0;
    waktu_off=0;
    set_off=0;
    delay_ms(100);
}

/*atur waktu jam*/

if ((PINC.0==0)&&(pilihset==2)&&(jamb==0))
{
    b=1;
    jamb=2;
}

if ((PINC.1==0)&&(a==0)&&(b==1))
{
    jamb=jamb-1;
    delay_ms(300);
}

if ((PINC.2==0)&&(a==0)&&(b==1))
{
    jamb=jamb+1;;
    delay_ms(300);
}

if ((b==1)&&(jamb==2))
{
    lcd_gotoxy(0,1);
    lcd_putsf("ATUR JAM    ");
    lcd_gotoxy(0,0);
    lcd_puts(cjam);
}

```



```

        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenit);
        lcd_gotoxy(5,0);
        lcd_putsf(":");
        lcd_gotoxy(6,0);
        lcd_puts(cdetik);
    }
    if ((b==1)&&(jamb==3))
    {
        lcd_gotoxy(0,1);
        lcd_putsf("ATUR MENIT  ");
        lcd_gotoxy(0,0);
        lcd_puts(cjam);
        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenit);
        lcd_gotoxy(5,0);
        lcd_putsf(":");
        lcd_gotoxy(6,0);
        lcd_puts(cdetik);
    }
    if ((b==1)&&(jamb==4))
    {

        lcd_gotoxy(0,1);
        lcd_putsf("ATUR DETIK  ");
        lcd_gotoxy(0,0);
        lcd_puts(cjam);
        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenit);
        lcd_gotoxy(5,0);
        lcd_putsf(":");
        lcd_gotoxy(6,0);
        lcd_puts(cdetik);
    }
    if ((b==1)&&(jamb==5))
    {

        lcd_gotoxy(0,1);
        lcd_putsf("BACK      ");
        lcd_gotoxy(0,0);

```

```

        lcd_puts(cjam);
        lcd_gotoxy(2,0);
        lcd_putsf(":");
        lcd_gotoxy(3,0);
        lcd_puts(cmenit);
        lcd_gotoxy(5,0);
        lcd_putsf(":");
        lcd_gotoxy(6,0);
        lcd_puts(cdetik);
    }
    if ((b==1)&&(jamb==6))
    {
        jamb=2;
        delay_ms(100);
    }
    if ((b==1)&&(jamb==1))
    {
        jamb=5;
        delay_ms(100);
    }

    if ((PINC.0==0)&&(pilihset==2)&&(jamb==5))
    {
        b=0;
        jamb=0;
        pilihset=1;
        pilihsetalat=0;
    }
    if ((PINC.4==1)&&(PINC.3==0))        //ok
    {
        if ((b==1)&&(jamb==2))    //ok, setting bertahap ++
        {
            jam=jam+1;    //ok
            if (jam==24)
            {
                jam=0;
                lcd_gotoxy(1,0);
                lcd_putsf(" ");
            }
            itoa(jam,cjam);
            lcd_gotoxy(0,0);
            lcd_puts(cjam);
        }

        if ((b==1)&&(jamb==3))    //ok, setting bertahap ++
        {

```

```

menit=menit+1; //ok
if (menit==60)
{
    menit=0;
    lcd_gotoxy(4,0);
    lcd_putsf(" ");
}
itoa(menit,cmenit);
lcd_gotoxy(3,0);
lcd_puts(cmenit);
}

if ((b==1)&&(jamb==4)) //ok, setting bertahap ++
{
    detik=detik+1; //ok
    if (detik==60)
    {
        detik=0;
        lcd_gotoxy(7,0);
        lcd_putsf(" ");
    }
    itoa(detik,cdetik);
    lcd_gotoxy(6,0);
    lcd_puts(cdetik);
}

if ((waktu_on==1)&&(set_on==2)) //ok, setting bertahap ++

{
    jamh=jamh+1; //ok
    if (jamh==24)
    {
        jamh=0;
        lcd_gotoxy(1,0);
        lcd_putsf(" ");
    }
    itoa(jamh,cjamh);
    lcd_gotoxy(0,0);
    lcd_puts(cjamh);
}

if ((waktu_on==1)&&(set_on==3)) //ok, setting bertahap ++
{
    menith=menith+1; //ok
    if (menith==60)
    {

```

```

        menith=0;
        lcd_gotoxy(4,0);
        lcd_putsf(" ");
    }
    itoa(menith,cmenith);
    lcd_gotoxy(3,0);
    lcd_puts(cmenith);
}

```

```

if ((waktu_on==1)&&(set_on==4)) //ok, setting bertahap ++

```

```

{
    jamh1=jamh1+1; //ok
    if (jamh1==24)
    {
        jamh1=0;
        lcd_gotoxy(1,0);
        lcd_putsf(" ");
    }
    itoa(jamh1,cjamh1);
    lcd_gotoxy(0,0);
    lcd_puts(cjamh1);
}

```

```

if ((waktu_on==1)&&(set_on==5)) //ok, setting bertahap ++

```

```

{
    menith1=menith1+1; //ok
    if (menith1==60)
    {
        menith1=0;
        lcd_gotoxy(4,0);
        lcd_putsf(" ");
    }
    itoa(menith1,cmenith1);
    lcd_gotoxy(3,0);
    lcd_puts(cmenith1);
}

```

```

if ((waktu_off==1)&&(set_off==2)) //ok, setting bertahap ++

```

```

{
    jamm=jamm+1; //ok
    if (jamm==24)
    {
        jamm=0;
    }
}

```

```

        lcd_gotoxy(1,0);
        lcd_putsf(" ");
    }
    itoa(jamm,cjamm);
    lcd_gotoxy(0,0);
    lcd_puts(cjamm);
}

if ((waktu_off==1)&&(set_off==3)) //ok, setting bertahap ++
{
    menitm=menitm+1; //ok
    if (menitm==60)
    {
        menitm=0;
        lcd_gotoxy(4,0);
        lcd_putsf(" ");
    }
    itoa(menitm,cmenitm);
    lcd_gotoxy(3,0);
    lcd_puts(cmenitm);
}

if ((waktu_off==1)&&(set_off==4)) //ok, setting bertahap ++

{
    jamm1=jamm1+1; //ok
    if (jamm1==24)
    {
        jamm1=0;
        lcd_gotoxy(1,0);
        lcd_putsf(" ");
    }
    itoa(jamm1,cjamm1);
    lcd_gotoxy(0,0);
    lcd_puts(cjamm1);
}

if ((waktu_off==1)&&(set_off==5)) //ok, setting bertahap ++
{
    menitm1=menitm1+1; //ok
    if (menitm1==60)
    {
        menitm1=0;
        lcd_gotoxy(4,0);
        lcd_putsf(" ");
    }
}

```

```

        itoa(menitm1,cmenitm1);
        lcd_gotoxy(3,0);
        lcd_puts(cmenitm1);
    }
    delay_ms(300);
}

if ((PINC.4==0)&&(PINC.3 ==1))
{
    if ((b==1)&&(jamb==2))  // setting bertahap --
    {
        jam=jam-1;  //ok
        if (jam== -1)
        {
            jam=23;
        }
        itoa(jam,cjam);
        if (jam<10)
        {
            lcd_gotoxy(1,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(0,0);
        lcd_puts(cjam);
    }

    if ((b==1)&&(jamb==3))  // setting bertahap --
    {
        menit=menit-1;  //ok
        if (menit== -1)
        {
            menit=59;
        }
        itoa(menit,cmenit);
        if (menit<10)
        {
            lcd_gotoxy(4,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(3,0);
        lcd_puts(cmenit);
    }

    if ((b==1)&&(jamb==4))  // setting bertahap --
    {
        detik=detik-1;  //ok

```

```

        if (detik==--1)
        {
            detik=59;
        }
        itoa(detik,cdetik);
        if (detik<10)
        {
            lcd_gotoxy(7,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(6,0);
        lcd_puts(cdetik);
    }

    if ((waktu_on==1)&&(set_on==2))    //ok, setting bertahap –

    {
        jamh=jamh-1;
        if (jamh==--1)
        {
            jamh=23;
        }
        itoa(jamh,cjamh);
        if (jamh<10)
        {
            lcd_gotoxy(1,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(0,0);
        lcd_puts(cjamh);
    }

    if ((waktu_on==1)&&(set_on==3))    // setting bertahap --
    {
        menith=menith-1;
        if (menith==--1)
        {
            menith=59;
        }
        itoa(menith,cmenith);
        if (menith<10)
        {
            lcd_gotoxy(4,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(3,0);
    }

```

```

    lcd_puts(cmenith);
}

if ((waktu_on==1)&&(set_on==4)) // setting bertahap --

{
    jamh1=jamh1-1; //ok
    if (jamh1==0)
    {
        jamh1=23;
    }
    itoa(jamh1,cjamh1);
    if (jamh1<10)
    {
        lcd_gotoxy(1,0);
        lcd_putsf(" ");
    }
    lcd_gotoxy(0,0);
    lcd_puts(cjamh1);
}

if ((waktu_on==1)&&(set_on==5)) // setting bertahap --
{
    menith1=menith1-1; //ok
    if (menith1==0)
    {
        menith1=59;
    }
    itoa(menith1,cmenith1);
    if (menith1<10)
    {
        lcd_gotoxy(4,0);
        lcd_putsf(" ");
    }
    lcd_gotoxy(3,0);
    lcd_puts(cmenith1);
}

if ((waktu_off==1)&&(set_off==2)) // setting bertahap --
{
    jamm=jamm-1; //ok
    if (jamm==0)
    {
        jamm=23;
    }
    itoa(jamm,cjamm);
}

```



```

        if (jamm<10)
        {
            lcd_gotoxy(1,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(0,0);
        lcd_puts(cjamm);
    }

    if ((waktu_off==1)&&(set_off==3))    // setting bertahap --
    {
        menitm=menitm-1;    //ok
        if (menitm==0)
        {
            menitm=59;
        }
        itoa(menitm,cmenitm);
        if (menitm<10)
        {
            lcd_gotoxy(4,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(3,0);
        lcd_puts(cmenitm);
    }

    if ((waktu_off==1)&&(set_off==4))    // setting bertahap --
    {
        jamm1=jamm1-1;    //ok
        if (jamm1==0)
        {
            jamm1=23;
        }
        itoa(jamm1,cjamm1);
        if (jamm1<10)
        {
            lcd_gotoxy(1,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(0,0);
        lcd_puts(cjamm1);
    }

    if ((waktu_off==1)&&(set_off==5))    // setting bertahap --
    {
        menitm1=menitm1-1;    //ok

```

```

        if (menitm1==1)
        {
            menitm1=59;
        }
        itoa(menitm1,cmenitm1);
        if (menitm1<10)
        {
            lcd_gotoxy(4,0);
            lcd_putsf(" ");
        }
        lcd_gotoxy(3,0);
        lcd_puts(cmenitm1);
    }

    delay_ms(300);
}

if
((PINC.1==1)&&(PINC.2==1)&&(PINC.3==1)&&(pilihset==1)&&(pilihsetalat==0))
{
    itoa(detik,cdetik);
    itoa(menit,cmenit);
    itoa(jam,cjam);
    lcd_gotoxy(5,0);
    lcd_putsf(":");
    lcd_gotoxy(6,0);
    lcd_puts(cdetik);
    lcd_gotoxy(2,0);
    lcd_putsf(":");
    lcd_gotoxy(3,0);
    lcd_puts(cmenit);
    lcd_gotoxy(0,0);
    lcd_puts(cjam);
    lcd_gotoxy(0,1);
    lcd_putsf("Kincir Oxygen ");
}
};
}

```